01311.001200 PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	Examiner: Q.N. Nguyen
RICHARD REISMAN	)	Group Art Unit: 2141
Application No.: 09/576,927	)	
Filed: May 23, 2000	)	
For: METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING	)	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131 OF RICHARD REISMAN

Sir:

The undersigned, RICHARD REISMAN, declares as follows:

1. I am the inventor of the invention described and claimed in the above-identified patent application. All acts hereunder took place in the United States.

2. Prior to January 14, 2000, I conceived the invention set forth in the pending claims of the subject patent application.

3. Evidence corroborating my conception of the claimed invention is found in four sheets of drawings that I created prior to January 14, 2000, copies of which are attached to a draft patent application (see paragraph 5 below) in Exhibit A. These four drawing sheets correspond to Figures 1B, 2, 3, 4A, 4B, and 5 of the subject patent application. My initials and a pre-January 14, 2000 date (redacted) are found on two of the four drawing sheets. However, I created all four drawing sheets prior to January 14, 2000

and provided them to my patent attorney, Joseph W. Ragusa of the law firm of Fitzpatrick, Cella, Harper & Scinto, also prior to January 14, 2000.

4. I understand that Mr. Ragusa added Figure 1A, the word "processing" to element 4 of Figure 1B, and the figure and reference numerals to all of the figures. But for these additions, the figures are identical to those I provided to him prior to January 14, 2000. With the above additions, figures 1A, 1B, 2, 3, 4A, 4B, and 5 are identical in substance to those of the subject patent application.

5. Prior to January 14, 2000, I received from Mr. Ragusa the draft patent application that he prepared, my original four drawing sheets having been modified by Mr. Ragusa as discussed above in paragraph 4. This draft patent application is attached as Exhibit A, and the pre-January 14, 2000 date thereon has been redacted. The draft application was accompanied by a cover letter from Mr. Ragusa, also attached in Exhibit A, which I also received prior to January 14, 2000 (the pre-January 14, 2000 date on this letter has been redacted, as well as any attorney-client privileged communications).

6. The draft patent application of Exhibit A, prepared by Mr. Ragusa, also corroborates my conception of the claimed invention prior to January 14, 2000.

7. Independent claims 6, 7, 13, 14, 20, 21, 27, 28 and 60-62, as per the Amendment filed on even date herewith, are fully supported by my original drawings, and by the pre-January 14, 2000 draft application discussed above (see, for example, Figs. 1B, 2, 4A and 4B and draft claims 1-21 of Exhibit A).

8. Further drafts of the patent application were reasonably diligently prepared by myself and Mr. Ragusa until the subject patent application was filed on May 23, 2000, which I understand constitutes a reduction to practice of the invention of the subject patent application. Copies of those further draft patent applications and the accompanying dated cover letters or e-mails are attached as Exhibits B-H (all attorney-client privileged communications have been redacted from these letters).

9. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of this application and any patent issuing thereon.

3/31/05

Date

Richard Reisman

NY_Main 491816_1.DOC

FITZPATRICK, CELLA, HARPER & SCINTO
30 ROCKEFELLER PLAZA
NEW YORK, NY 10112-3801
212-218-2100
FACSIMILE (212) 218-2200

WASHINGTON OFFICE
1900 K STREET, N.W.
WASHINGTON, D.C. 20006-1110
(202) 530-1010
FACSIMILE (202) 530-1055

CALIFORNIA OFFICE
650 TOWN CENTER DRIVE, SUITE 1800
COSTA MESA, CALIFORNIA 92626-1925
(714) 540-8700
FACSIMILE (714) 540-9823

WRITER'S DIRECT DIAL NUMBER

REDACTED

VIA FEDERAL EXPRESS

Mr. Richard Reisman
20 East 9th Street
Apt. 14K
New York, NY 10003

Re: New Application for Community Usage Learning

Dear Dick:

REDACTED

I enclose a draft specification, claims and drawings for your invention referred to in your faxes as invention 3c (signifier mapping with CUL).

REDACTED

Sincerely yours,

Joe

Joseph W. Ragusa

JWR:jrl
Enclosure

REDACTED

TITLE

METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is directed to a computer-implemented product for locating and connecting to a particular desired object or target resource from among plural resources resident at distributed locations on a network.

Description of the Related Art

The worldwide network of computers known as the Internet evolved from military and educational networks developed in the late 1960's. Public interest in the Internet has increased of late due to the development of the World Wide Web (hereinafter, the Web), a subset of the Internet that includes all connected servers offering access to hypertext transfer protocol (HTTP)

space. To navigate the Web, browsers have been developed that give a user the ability to download files from Web pages, data files on server electronic systems, written in HyperText Mark-Up Language (HTML). Web pages may be located on the Web by means of their electronic addresses, known as Uniform Resource Locators (URLs).

A URL uniquely identifies the location of a resource (web page) within the Web. Each URL consists of a string of characters defining the type of protocol needed to access the resource (e.g., HTTP), a network domain identifier, identification of the particular computer on which the resource is located, and directory path information within the computer's file structure. The domain name is assigned by Network Solutions Registration Services after completion of a registration process.

While the amount of information available on the Web is enormous, and therefore potentially of great value, the sheer size of the Web makes the search for information, and particular web sites or pages, a daunting task. Search engines have been developed to assist persons using the Web in searching for web pages that may contain useful information.

Search engines fall into two major categories. In search engines falling into the first category, a service provider compiles a directory of Web sites that the provider's editors believe would be of interest to users of the service. The Yahoo site is the best known example of such a provider. Products in this category

are not, strictly speaking, search engines, but directories, and will be referred to hereinafter as "editor-controlled directories". In an editor-controlled directory, the developer of the directory (the "editor") determines, based upon what it believes users want, what search terms map to what web pages.

The other major category, exemplified by Altavista, Lycos, and Hotbot, uses search programs, called "web crawlers", "web spiders", or "robots", to actively search the Web looking for web sites containing information relevant to a requested search. The web crawler compares the terms in a search request with information resident in millions of available Web documents. Most commonly, this is done by examining descriptive information about the Web document contained in a tag or tags in the header of a page. Such tags are known as "metatags" and the descriptive information contained therein as "metadata". These products will be referred to hereinafter as "author-controlled search engines," since the authors of the Web documents themselves control, to some extent, whether or not a search will find their document, base upon the metadata that the author includes in the document.

Each type of product has its disadvantages. Author-controlled search engines tend to produce search results of enormous size. However, they have not been reliable in reducing the large body of information to a manageable set of relevant results. Further, web site authors often attempt to skew their site's position in the search results of author-controlled search engines

by loading their web site metatags with multiple occurrences of certain words commonly used in searches.

Editor-controlled directories are more selective in this regard. However, because conventional editor-controlled directories do not actively search the web for matches to particular search terms, they may miss highly relevant web sites that were not deemed by the editors to be worthy of inclusion in the directory. Also, it is possible for the editor to "play favorites" among the multitude of Web documents by mapping certain Web documents to more search terms than others.

Recently, search engines such as DirectHit (www.directhit.com) have introduced feedback and learning techniques to increase the relevancy of search results. DirectHit purports to use feedback to iteratively modify search result rankings based on which search result links are actually accessed by users. Another factor purportedly used in the DirectHit service in weighting the results is the amount of time the user spends at the linked site. The theory behind such techniques is that, in general, the more people that link on a search result, and the longer the amount of time they spend there, the greater the likelihood that users have found this particular site relevant to the entered search terms. Accordingly, such popular sites are weighted and appear higher in subsequent result lists for the same search terms. The Lycos search engine (www.lycos.com) also uses feedback to weight its search results. In the Lycos search engine, as described in U.S. Patent No.

5,748,954, weight is allocated based upon how many times a listed web site is linked to from other web sites.

Even leaving aside the drawbacks discussed above, search engines of both categories are only useful when a user desires a list of relevant web sites for particular search terms. Often, users wish to locate a particular web site but do not know the exact URL of the desired web site. Conventional search engines are not the most efficient tools for doing this.

Moreover, naming and locating particular sites on the Web is currently subject to serious problems. For example, appropriate names, including existing company names or trademarks, may not be available, because someone registered them first. Names may be awkward and not obvious, because of length, form/coding difficulties or variant forms, and names may not justify a separate domain name registration for reasons of cost and convenience, such as movie titles or individual products.

This problem results from a mismatch between the present network addressing scheme based on Uniform Resource Locators (URLs), which meet the technical needs of the Internet software, and the needs of human users and site sponsors for simple, user-friendly mnemonic and branded names. This problem is largely hidden in cases where a user finds a site by clicking a pre-coded link (such as after using a search engine), or by using a saved bookmark. However, the problem does seriously affect users wishing to find a site

directly, or to tell another person how to find it. To do this, the person must know and type the URL into his Internet browser, typically of the form sitename.com or www.sitename.com. Site sponsors are also seriously hampered by this difficulty in publicizing their sites.

Further, the current method of naming and locating Web sites has serious, widely known problems. Web site locator "domain" names are often not simple or easily remembered or guessed, and often do not correspond to company, trademark, brand or other well-known names.

As a result of the foregoing, site URLs (or domain names) are not intuitively obvious in most cases, and incorrect access attempts waste time and produce cryptic error messages that provide no clue as to what the correct URL might be. A significant percentage of searches are for specific, well-known sites. These could be found much more quickly by a special-purpose locator engine. The current mode of interacting with search engines is also cumbersome-for this purpose, a much simplified mode of direct entry is practical.

One attempt to provide the ability to map a signifier, or alias, to a specific URL utilizes registration of key words, or aliases, which when entered at a specified search engine, will associate the entered key word with the URL of the registered site. One such commercial implementation of this technique is known as NetWord (www.netword.com). However, the NetWord aliases are assigned on a registration basis, that is, owners of web sites pay NetWord a registration fee to be mapped to by a particular key word. As a result,

the URL returned by NetWord may have little or no relation to what a user actually would be looking for. Another key word system, RealNames (www.realnames.com), similarly allows web site owners to register, for a fee, one or more "RealNames" that can be typed into browser incorporating RealNames' software, in lieu of a URL. Since RealNames also is registration based, there is no guarantee that the URL to which is user is directed will be the one he intended.

However, none of the currently available key word systems utilize heuristic techniques actually to determine the site intended by the user. Instead, the current systems teach away from such an approach by their use of registration, rather than user intention, to assign key words to map to web pages. Thus, the current techniques are not directed to solving the problem of finding the one, correct site for a particular signifier.

Thus, the need exists for a system that would enable a user to find a desired Web document by simply entering an intuitive key word or alias and that would perform a one to one mapping of the alias with the URL actually desired by the user, and which would use heuristic techniques to assist in providing the correct mapping, and improving system accuracy over time.

SUMMARY OF THE INVENTION

In consideration of the above deficiencies of the prior art, it is an object of the present invention to provide a method of signifier mapping that allows a

user to locate to a particular network resource, in the preferred embodiment a web page, by simply entering a signifier or alias.

It is a further object of the present invention to provide a system in which heuristic techniques are used together with user feedback to improve the accuracy of signifier mapping.

The present invention has several advantageous features, various combinations of which are possible:

1) a special purpose mapping engine for locating popular sites by guessed names;
2) automatic display of the target site (if located with reasonable confidence);
3) an optional simplified mode of direct entry of a guessed site name; and
4) use of user expectations, such as popularity of guesses intended for a given site, as a primary criterion for translating names to sites, with provision for protection of registered trademarks or other mandates.

In accordance with one aspect of the present invention, a finder or locator server is established. The server is configured to generate a user interface that allows users to enter an guessed name or alias, as easily as if the user knew the correct URL for the intended target resource. In response to entry of the alias, the finder server accesses a database that includes, in a preferred embodiment, domain name registration information, as well as user feedback from previous users of the server, and looks up the correct URL,

i.e., the one URL that corresponds to the alias and causes the user's browser to go automatically to that URL, without the user having to view and click on a search results page, if the correct URL can be determined with a predetermined degree of confidence.

In one preferred embodiment, the server is structured to accept the alias as a search argument and do a lookup of the correct URL and the return of same to the browser, without the intermediate step of the user having to visit a search web page. The automatic transfer is preferably effected using standard HTML facilities, such as a redirect page or framing. Redirect is effected by placing pre-set redirection pages at the guess URL on the server. Alternately, the redirect page can be generated dynamically **[HOW?]**

The present invention advantageously uses feedback and heuristic techniques to improve the accuracy of the determination of the correct URL. If an exact match is found by the look-up technique and the accuracy of the mapping is confirmed by user feedback, then, after directing the user to the URL, the result is stored in the server to improve the accuracy of subsequent queries. The server database includes a list of expected terms and expected variants that can initially be catalogued to provide for exact matches. This list is updated by the learning processes discussed in more detail below.

If, on the other hand, an exact match cannot be determined, the finder server preferably uses intelligent techniques to find a probable match,

together with a selection of links to possible matches, or could return a no-match page with advice, or a conventional search interface or further directories.

According to a preferred embodiment of the invention, each of the selection of links are configured not to go directly to the target URL. Rather, the links are directed back to a redirect server established by the finder server, with coding that specifies the true target, and feedback information. The finder server can in this way keep track of user selections.

In accordance with an advantageous aspect of the invention, such feedback information is used to improve the results of the search by promoting web sites almost universally selected to exact match status, and by improving the ranking of possible lists in accordance with which links are most often selected. Preferably, a confidence parameter can be generated from such tracking to control whether to redirect to a URL or to present a possible list to users.

In furtherance of the above and other objects, there is provided, a designated server, accessible on the Internet, the designated server being configured to respond to relocation requests that specify an identifier, corresponding to a target resource, that may not be directly resolvable by standard Internet Protocol name resolution services to the URL of the target resource. In a direct entry embodiment of the present invention, requests are passed to the relocation server by sending a relocation URL that designates the relocation server as the destination

mode and appends the identifying information for the identifier as part of a URL string. The relocation server extracts the identifying information and translates it into a valid URL for the target resource. The relocation server is configured, in the event that a unique URL can be determined with respect to the target resource, to cause the target resource to be presented to the user without further action on the part of the user.

Preferably, the user requests are entered at a web browser, the response causes the target resource to be obtained by the user's web browser, and the relocation server determines the valid URL for the target resource by performing a look-up in a database.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A illustrates the finder server of the present invention connected to users and other resources via the Internet;

Figure 1B is a flow diagram illustrating a method of obtaining feedback from multiple users to be applied in searching or signifier mapping;

Figure 2 is a flow diagram showing a method of signifier mapping using feedback and heuristics to continually improve the performance of the mapping;

Figure 3 shows an example of a database entry for the finder server of the present invention;

Figure 4A is a flow diagram illustrating a technique of feedback weighting for probable results in signifier mapping;

Figure 4B is a flow diagram illustrating a technique of feedback weighting for possible results in signifier mapping; and

Figure 5 is a flow diagram illustrating how feedback is used in a preferred embodiment to discriminate a probable target resource in accordance with the present invention.

DETAILED DESCRIPTION OF THE DISCLOSED EMBODIMENTS

The present invention relates to a method and apparatus for locating a desired target resource located and accessible on a network, in response to user entry of a guessed name or alias. In illustrating the preferred embodiment, the apparatus is shown as a server computer, or computers, located as a node on the Internet. However, the present invention is in no way limited to use on the Internet and will be useful on any network having addressable resources.

The finder server of the preferred embodiment of the present invention allows users to enter a guessed identifier or alias, as easily as if they knew the correct URL. Specifically, the finder server of the present invention accepts a guessed name, or alias, from a user, uses a look-up technique, enhanced by heuristics preferably taking into account previous users' actions, to determine a correct URL for the

intended target resource, and causes the user's browser to go to that URL automatically. Preferably this is done without the added step of first viewing and clicking on a search-results page, where an initial search finds the intended target resource with a predetermined degree of certainty. Such a resource will be referred to hereinafter as a "probable". In accordance with a preferred embodiment of the present invention, this functionality can be implemented by:

- Publicizing the locator server under an appropriate URL name, for example, guessfinder.com.
- Setting up the server to, in response to entry of a guessed name or alias, do a lookup to the correct URL and return a response that causes the user's browser to go automatically to the specified URL. Such an automatic transfer can be effected using a standard HTML facilities, such as a redirect page, or framing.
- If the guess does not provide an exact match in the lookup phase, using feedback and heuristic techniques to create and present to the user a selection of links to possible matches. Alternately, the user may be presented with a nomatch page with advice, or directed to a conventional search interface, or further directories.

It is contemplated that the use of aliases for attempting to locate a web site associated with company name or brand name would be found useful. For example, the aliases "s&p", "s-p", "sandp", "snp",

"standardandpoors", "standardnpoors", "standardpoors" should preferably all map to www.standardpoors.com. In addition to companies and brands, other important name domains would include publications, music groups, sports teams, and TV shows.

The present invention advantageously provides for learning and feedback on the basis of user preferences to automatically and dynamically build a directory of names and sites that maps to the actual expectations and intentions of a large population of users, and adapts to changes over time, including the appearance of new sites, thus optimizing utility to them.

The finder server of the present invention effectively provides a secondary name space, administered by the organization operating the finder system, that maps to, but is not dependent on, the URL name space. The finder site computer has access to a data base containing entries for any number of popular sites, with any number of likely guesses and variations for each site.

As a result of the service provided when the present invention is implemented, site sponsors could skip the cumbersome and costly process of obtaining specific mnemonic URLs or alternate URLs in many cases (especially with regard to domain names). Even with a number of conventional URLS, this service could be a supplement, for additional variations. The problem of pre-empted URL domain names would also be avoided, except where there is legitimate and significant pre-existing usage.

Figure 1A illustrates a first embodiment of the present invention, as implemented on the Internet. The finder server 10 includes a computer or computers that perform processing, communication, and data storage to implement the finder service. Finder server 10 includes a finder processing/learning module 101. Module 101 performs various processing functions, and includes a communication interface to transmit and receive to and from the Internet 12, as well as with database 102, and is programmed to be operable to learn from experiential feedback data by executing heuristic algorithms. Database 102 stores, in a preferred embodiment, indexes of URL data that would allow the module 101 to locate, with a high degree of confidence, a URL on the Web that is an exact match for a target resource in response to a user's entry of an alias or guessed name. Preferably, the indexes store, in addition to available URL information, such as domain name directories, information relating to the experience of the server in previous executions of the finder service. As the server gains experience and user feedback, heuristic techniques are applied module 101 to enable the returned URLs to conform more and more accurately to user expectations.

User interface/browser 11 is located on the user's computer. As has been discussed previously, the user enters a guessed name, or alias, into the browser and submits a query containing the alias to the finder server. Computers 14, 15 . . . n represent computers connected to the Internet, each having associated therewith one or more URLs, each of which forming the address of a target resource.

The finder server includes operating system servers for external communications with the Internet and with resources accessible over the Internet. Although the present invention is particularly useful in mapping to Internet resources, as was discussed above, the method and apparatus of the present invention can be utilized with any network having distributed resources.

Entry of the alias by the user may be accomplished in a number of ways. In one embodiment, a usage convention can be publicized for passing the alias to the server within a URL string, such as guessfinder.com/ibm, for example, for trying to find the web page corresponding to the alias "ibm". The server is programmed to treat the string "ibm" as a search argument and perform the appropriate processing to map the alias to the intended target resource. Alternately, the user can visit the web site of the finder server and download a search form, as is done in conventional search engines. A third option is to provide a browser plug in that allows direct entry of the key word in the browser's URL window.

It also would be preferable for an enhanced user interface to be phased in as the service gains popularity. This preferably would be accomplished by a browser plug-in, or modifications to the browser itself, to allow the alias to be typed into the URL entry box without need for the service domain name prefix (such as, guessfinder.com/...). Instead, such an entry would be recognized as a alias, not a URL, and the prefix would be appended automatically, just as

http://... is appended if not entered with a URL in current browsers.

Figure 1B is a flow diagram illustrating a technique for obtaining and learning from feedback responses gathered from a large group of people, in the example, users 1, 2, . . . n. Such a technique can be used in a variety of applications, and in particular in traditional search engines, or in mapping to identify particular web sites, as in alias or signifier mapping.

In Figure 1B, users 1, 2, . . . n represent a large community of users. In the flow diagram, the flow of query items from the users is indicated by a Q, the flow of responses back to the users is indicated by an R, and the flow of feedback results provided by the users' actions, or responses to inquiries, is indicated by an F. As can be seen from the figure, Query (a, 1) is transmitted from user 1 to the service 2, which can either be a searching or a mapping service. The service has learning processor 4, which interfaces with a database 6. The database 6 contains, among other things, indexes and feedback information gathered from previous queries. In response to the query, the user 1 is provided with a response R(a, 1). User 1 then is provided with the opportunity to transmit user Feedback (a, 1) to the Service 2. Learning processor 4 stores the feedback information in the database 6, and is programmed with one or more heuristic algorithms enabling it to learn from the feedback information to improve the returned search or mapping results. The feedback provided will improve the results offered, for example by positively weighting results preferred by

users, so that, over time, more accurate results can be obtained.

Figure 2 is a diagram illustrating the logical flow used in applying the general technique of learning from user feedback shown in Figure 1 to signifier mapping, in accordance with a preferred embodiment of the present invention. A user enters a Query consisting of a signifier, represented by Q^s. The server, in response to receipt of the query, parses the query, at step S02, and in step S04 performs a database lookup in an attempt to determine, if possible, the exact target resource intended by the user. Database 6 includes index data as well as feedback data obtained from users in previous iterations of the signifier mapping program, is accessed. The stored data structure is described in more detail below.

In step S06, the program discriminates a probable intended target making use of the index information such as domain registration indexes, and other resources, as well as the feedback information stored in the database. In step S08, if a likely hit, or exact match has been identified, that is, a web page has been located with a high confidence parameter, the flow continues to step S10. At step S10, a direction is prepared to the likely hit URL. A list of alternatives optionally may be provided for presentation to the user at the same time, in case the likely hit turns out not to be the target identifier. At step S12, the server sends information R^s to the user, more particularly to the user's browser, to

effect a link to the likely hit. Optionally, the alternate list is also provided at the same.

In step S14, the viewed page is monitored by the server and the user, by his actions, provides feedback. For example, in addition to the fact of the user having chosen the link, the amount of time the user spends at the site will be an indicator of whether the site is the intended target. Other feedback can be provided by asking the user. The URL of the viewed page is recorded, together with any other feedback, for use in improving the accuracy of subsequent iterations of signifier mapping. At step S26, the feedback data is supplied to a feedback weighting algorithm, described in detail below, which generates appropriate weighting factors to be stored in the database for use in subsequent mappings.

If it is determined at step S08 that the result is not a likely hit, the flow proceeds to step S18, where a list of the top m hits (m being a predetermined cutoff number), preferably from a search engine, is prepared. The list is presented, at step S20, to the user as R^s. The user, by the selections made from the provided list, and from other feedback, such as how long the user spends at each link, supplies feedback to the system. This information F^s is monitored, at step S22 and recorded, at step S24. The recorded information is supplied to the feedback weighting algorithm, at step S26, the output of which is stored in the database for use in subsequent iterations of the signifier mapping.

Figure 2 illustrates the simple case in which a user is directed to a target URL if the target has been determined to be a probable hit, and is presented with a list to choose from if the target cannot be identified with sufficient certainty. However, it is well within the intended scope of the invention for alternate methods to be employed. For example, the user interface (UI) could be extended, either by a browser plug in, or at a dedicated search site, to provide multi-pane/multi-window results that allow a pane for each type of response, e.g., the target response and a list of possibles, regardless of the level of confidence in the result. In such a case, the format for presentation of results would be the same whether a probable has been located or not, but the learning from feedback and ranking would still seek to determines "correctness" based on the varying feedback cases.

Figure 3 illustrates a preferred method of organizing index data to allow for storing and updating of the most probable hits for a given query. As can be seen from the illustration, for each query, whether single element queries or compound queries, there is stored a list of associated possible targets. Linked to each of these query/target pairs is a raw score, an experience level, and a probability factor. As feedback enters the system, the index data is updated to reflect the user feedback. The updating process will be described below. While the index shows preferred weighting criteria, these are only a sample of the kind of criteria that can be correlated to the query/target pairs. **{please supply examples of others}**

Figure 4A illustrates a preferred technique for weighting the results using feedback data for hits that have been determined to be probable hits. In step S30, if the user feedback from the probable result indicates that the probable was in fact the target URL the user was searching for, the flow proceeds to step S32 where the raw score for that query/target pair is incremented by $factor_Y$. If the user returns feedback indicating that the probable was not the target resource the user had in mind, the flow proceeds to step S34 where the raw score for that query/target pair is decremented by $factor_N$. If the user provides no feedback, then the flow proceeds to step S36 where the raw score is decremented by $factor_O$, which can be zero. After execution of any of steps S32, S34 or S36, the flow proceeds to step S38, at which the experience level score is incremented by $Efactor_C$.

Figure 4B illustrates a preferred technique for weighting in accordance with user feedback in the case of possibles, i.e., items on the list presented to the user when no probable result can be located. As shown in the figure, if a possible is selected by the user from the presented list, at step S40, the fact of selection is recognized, preferably by use of a redirect server that allows the system to keep track of which link was chosen. Additionally, the amount of time the user spends at the selected link is ascertained. Making use of the information gathered in the redirect, the raw score for the query/target pair is incremented, at step S44, by $factor_S$. The user is then requested to provide additional feedback after the user has finished viewing the link.

In a preferred embodiment of the present invention, this feedback is gathered from the user by presenting the user with a frame that includes a mechanism, such as a check box, or radio button, that allows the user to indicate whether the selected possible was in fact the intended or "correct" target resource. If it is determined, at step S42, from the feedback that the link was the correct target, the flow proceeds to step S46, where the raw score for that query/target pair is incremented by $\text{factor}_{Y'}$. If the user returns a negative response, the raw score of the pair is decremented at step S48 by a by $\text{factor}_{N'}$. If no feedback is received, the raw score is decremented, at step S50, by $\text{factor}_{O'}$, which can be zero. After execution of any of steps S44, S46, S48 or S50, the flow proceeds to step S38, at which the experience level score is incremented by Efactor_{PS} in the case of selection of the link, and by Efactor_{PC} if the link was the correct. **{I'm not sure this is correct. Please check the flow}.**

Figure 5 illustrates a detail of how the present invention ranks and discriminates a probable target. At step S100 a list of possibles is obtained. Next, the list is ranked, at step S102, on the basis of the expected probability as the target. In step S102, a discrimination criteria is calculated and compared with a predetermined threshold parameter. For example, **{The formula in the flow chart does not define what the Probabilities T1 and T2 represent. Please amplify this portion of the description.}**

In the preferred embodiment, when a link on a list of possibles is selected by the user, rather than connect the user immediately to the chosen link, the finder server first redirects the user to a redirect server where feedback data relating to the selection can be gathered. One item of feedback is the very fact of the selection. Further feedback can be obtained by monitoring how long the user spends at the selected link, and by directly querying the user.

The linking technique uses the URL as a server parameter within a URL to control the intermediate server parameter within the URL to control the intermediate server process. Thus a server is called with a first URL that specifies the second URL as a parameter. For example

http://archive.com/relocator?server.com/path1/page1.htm

where archive.com is the intermediate server URL, and server1.com/path1/page1.htm is the target. The network ignores the parameter portion of the URL, which is passed as data to the server. The server acts on the parameter to perform desired intermediary processing, and to redirect the user to the intended location specified by the second URL. A more advanced form of usage could impose limited added structure, to add precision and avoid ambiguities, such as guessfinder.com/movie/amistad, to specify that the alias refers to the movie Amistad.

The delay required for the redirect provides the opportunity for the display of interstitial advertisements. In addition, additional user feedback can be solicited during the delay, and the connection to the targeted URL can be aborted if the user indicates that the target site is not the one he or she intended. In addition to using the redirect when a link is selected, the technique also preferably is used when an exact match is found, to provide a brief delay before connecting the user to the exact match, to present advertisements to give the user the time to abort the connection. In any event, the user preferably is given the opportunity to provide feedback after connecting to any site, whether directly as a result of an exact match, or as a result of selecting from a linked possibles list.

The redirect server of the present invention allows data to be gathered on each link as it is followed and redirected. The redirect link can be created in a simple static HTML. However, it is preferable to create the link dynamically for each user selection.

The above embodiments of the present invention have been described for purposes of illustrating how the invention may be made and used. However, it should be understood that the present invention is not limited to the illustrated embodiments and that other variations and modifications of the invention and its various aspects will become apparent, after having read this disclosure, to those skilled in the art, all such variations and modifications being contemplated as

falling within the scope of the invention, which is defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method of finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained on the database, the method comprising:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

2. A method according to Claim 1, wherein a resource is determined at the accessing step to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

3. A method according to Claim 2, wherein if the guessed name does not map uniquely to any of the

indexed resources, to at least the predetermined confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

4. A method according to Claim 3, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

5. A method according to Claim 1, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

6. An apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database;

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if

any, of the resource determined to be the target resource.

7. An apparatus according to Claim 6, wherein a resource is determined to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

8. An apparatus according to Claim 7, wherein the apparatus is operable to, if the guessed name does not map uniquely to any of the indexed resources, to a predetermined confidence level, perform the following steps:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

request that the user select a link; and

cause the user's computer to go to the selected link, if any.

9. An apparatus according to Claim 8, wherein the apparatus is operable to, if a link has been selected, perform the following steps:

add information regarding the selection of the link to the feedback information in the database;

solicit user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, increment the confidence factor associated with the mapping between the guessed identifier and the address of the

selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

10. An apparatus according to Claim 6, the apparatus being further operable to:

solicit user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, increment the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

11. A system for finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the system comprising:

finder server means having: (a) database means for storing an index of resources available on the

network; and information regarding user feedback gathered in previous executions of the system; and (b) learning system means for accessing and learning from information contained on the database;

receiving means for receiving a guessed resource name from the user;

accessing means for accessing the database means to determine which, if any, of the indexed resources is the desired target resource;

directing means for directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined in the access means to be the target resource.

12. A system according to Claim 11, wherein a resource is determined by the access means to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

13. A system according to Claim 12, further comprising:

presenting means for, if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting means for requesting that the user select a link from the presented list; and

means for causing the user's computer to go to the selected link, if any, upon a user selection of a link.

14. A system according to Claim 13, further comprising:

adding means for, if a link has been selected, adding information regarding the selection of the link to the feedback information in the database;

soliciting means for soliciting user feedback with regard to the selected link; and

means for, if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

15. A system according to Claim 11, further comprising:

soliciting means for soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

means for, if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the

resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

16. A computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method comprising:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

17. A computer-readable medium according to Claim 16, wherein a resource is determined in the accessing step to be the target resource if the database information indicates that there is at least a predetermined

confidence level that the guessed name maps uniquely to the address of the resource.

18. A computer-readable medium according to Claim 17, wherein if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

19. A computer-readable medium according to Claim 18, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

20. A computer-readable medium according to Claim 16, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

21. A system for finding resources on a network of interconnected computers on which a plurality of resources reside, the system comprising:

a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and

a finder server having:

(a) a database including: (i) an index of resources available on the network; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and

(b) a learning system operable to access and learn from information contained on the database,

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

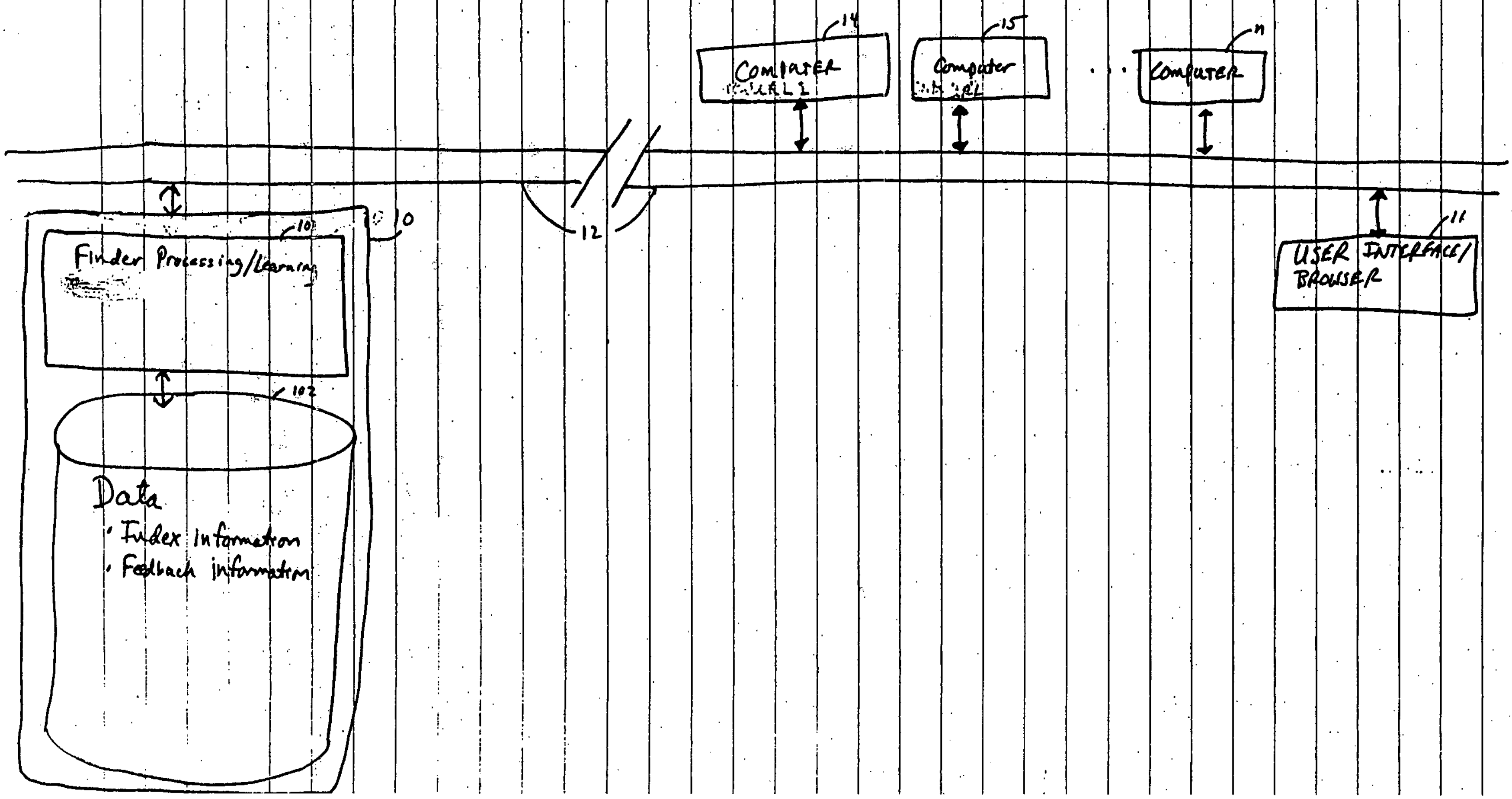
directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

ABSTRACT OF THE DISCLOSURE

An apparatus for finding resources on a network comprises a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database. The finder server is operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by: receiving a guessed resource name from the user; accessing the database to determine which, if any, of the indexed resources is the desired target resource; directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

NY_MAIN 50188 v 1

Fig. 1A.



COMPUTER
Computer
Computer
14
15
n
12
10
101
102
Finder Processing/Learning
Data
· Index information
· Feedback information
USER INTERFACE/
BROWSER
11

Generic CUL
Multi-user feedback

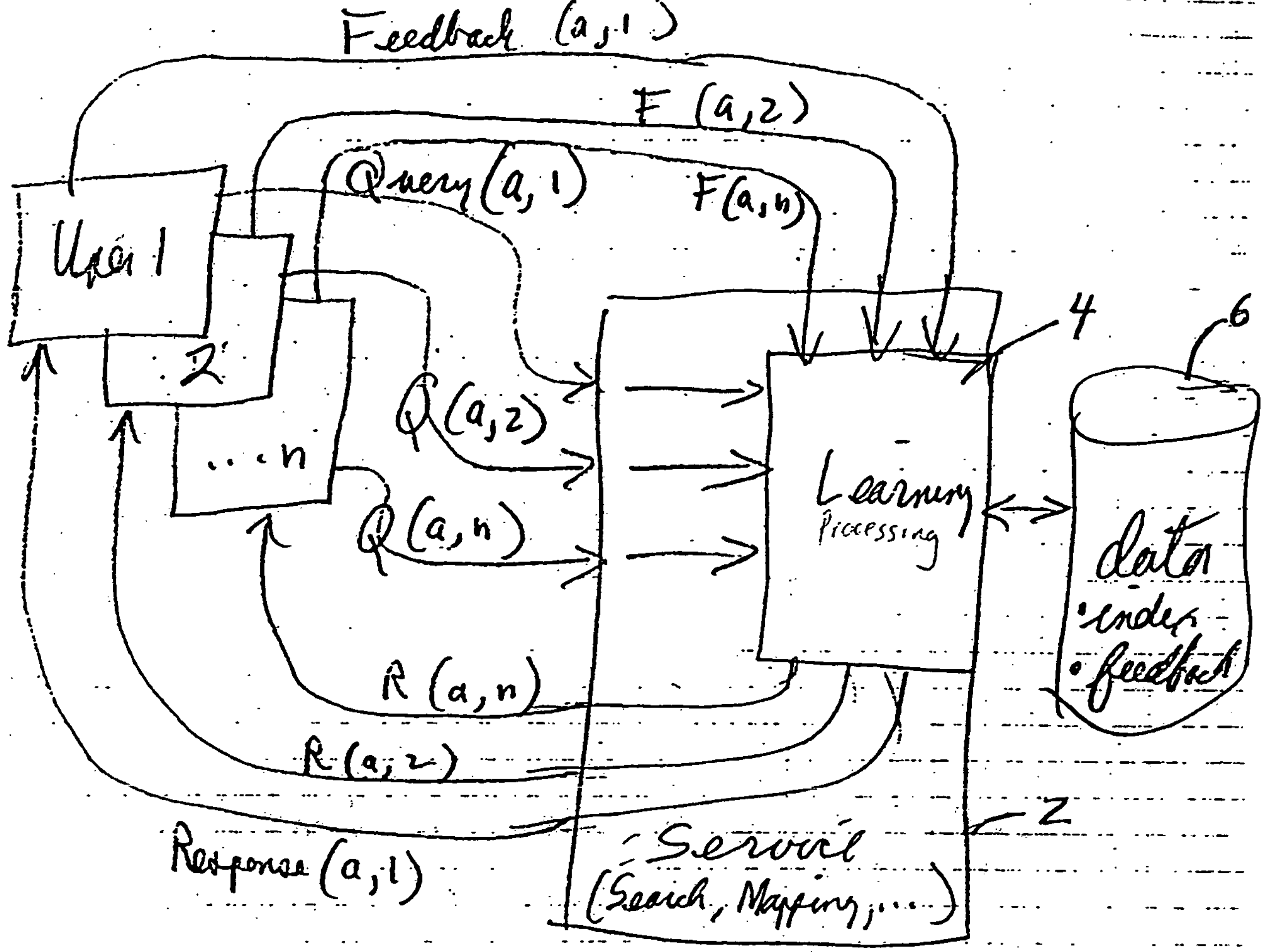


Fig. 1B

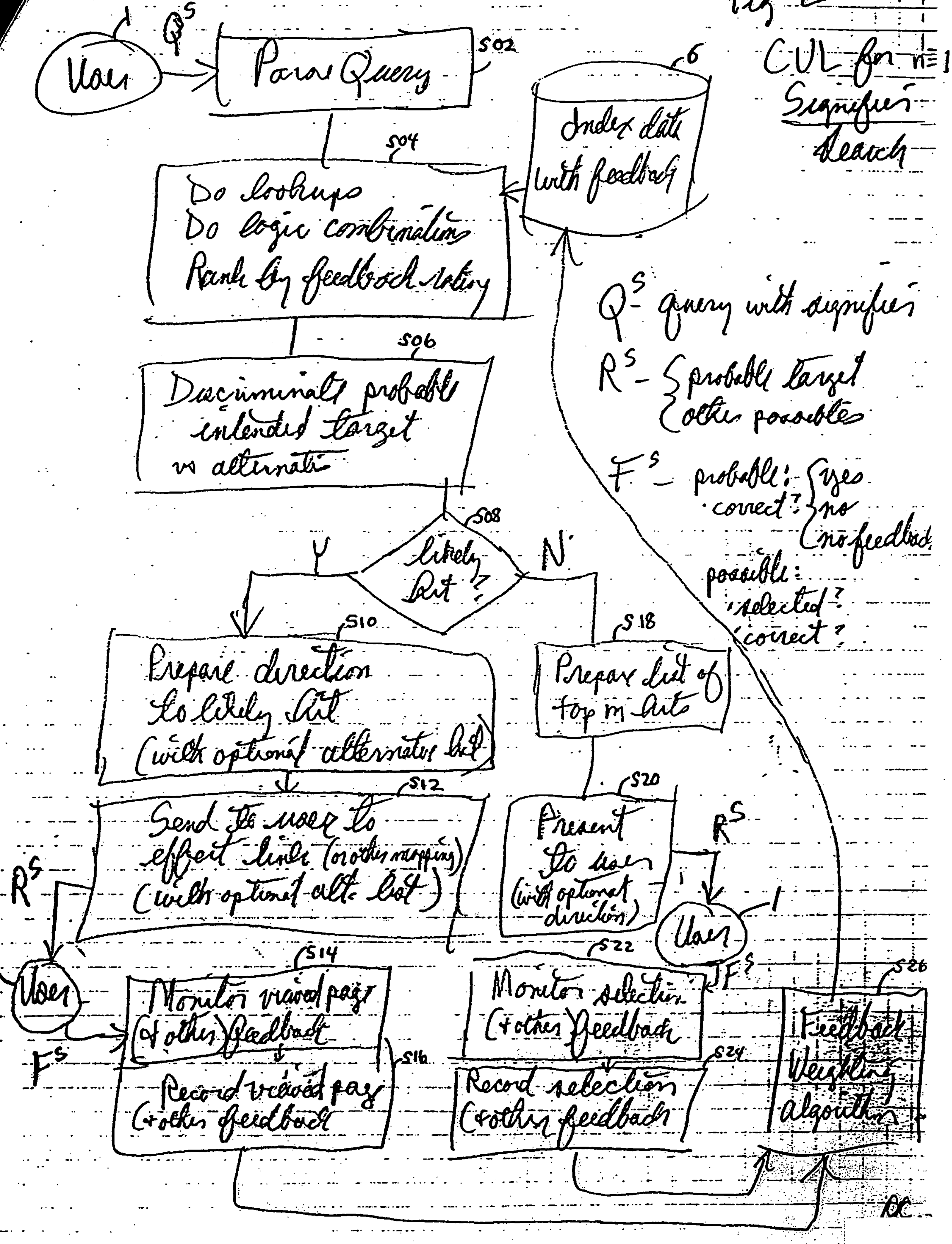
Q, R, F (a, n) (Query or request item, user case/instance) = Query item; = Query response; = Feedback results

RC

REDACTED

Fig 2

CUL for n=1 Signifier Search



REDACTED

Sample CUL Signifier Fig. 3

Feedback weighting/ranking (many variations/enhancement

Index data – selected elements

Single element Q's	Possible Target	Raw score	Experience level	Probability factor
Q (a)	1	—	—	—
	2	—	—	—
	3	—	—	—
Q (b)	1	—	—	—
	⋮	—	—	—
Compound/(Multi-element) Q's				
Q (c)	1			
	2			
	⋮			

Feedback Weighting – detail

Fig. 4A → For probables –

S530 Correct?
— yes → S532 increment raw score by $factor_Y$
— no → S534 decrement raw score by $factor_N$
— no feedback → S536 decrement raw score by $factor_0$ (can be = 0)

S538 for all, increase Experience level score by $E\ factor_C$ →

Fig. 4B → For possibles

S540 Selected? — Y → S544 increment raw score by $factor_S$
N ←

S542 Correct? — Y → S546 increment raw score by $factor_Y$
N ← S548 Decrement Raw Score by $factor_N$
No Feedback → S550 decrement raw score by $factor_0$ (may be zero)

S552 increase experience level by $E\ Factor_{PS}$, $E\ Factor_{PC}$ →

Sample (p 2)

Rank + Discriminate Probable Target – detail

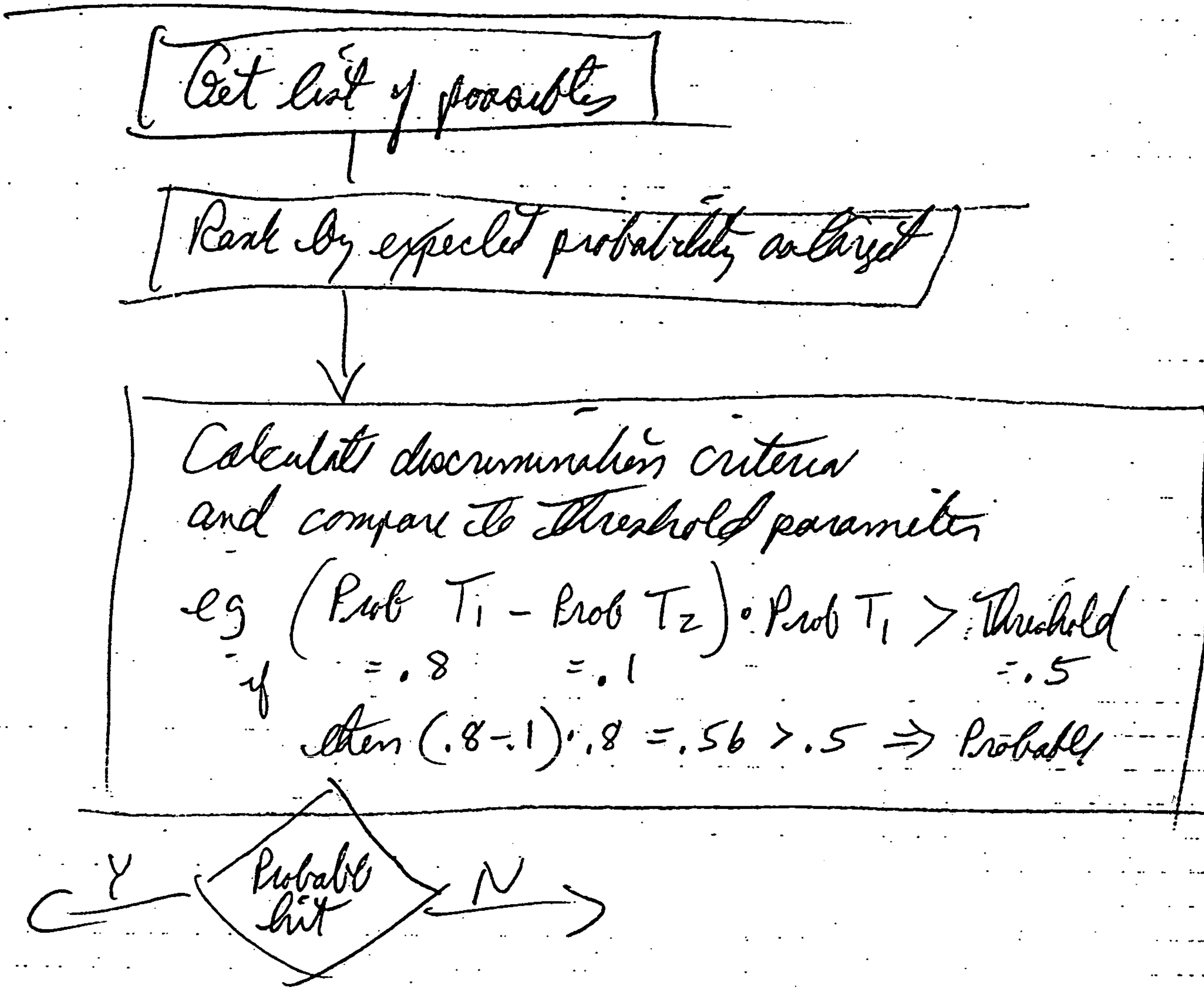


Fig. 5

FITZPATRICK, CELLA, HARPER & SCINTO
30 ROCKEFELLER PLAZA
NEW YORK, NY 10112-3801
212-218-2100
FACSIMILE (212) 218-2200

WASHINGTON OFFICE
1900 K STREET, N.W.
WASHINGTON, D.C. 20006-1110
(202) 530-1010
FACSIMILE (202) 530-1055

CALIFORNIA OFFICE
650 TOWN CENTER DRIVE, SUITE 1800
COSTA MESA, CALIFORNIA 92626-1925
(714) 540-8700
FACSIMILE (714) 540-9823

WRITER'S DIRECT DIAL NUMBER
(212) 218-2223

February 8, 2000

REDACTED

VIA FEDERAL EXPRESS

Mr. Richard Reisman
20 East 9th Street
Apt. 14K
New York, NY 10003

Re: New Application for Community Usage Learning

Dear Dick:

Enclosed is a copy of the Community Usage Learning application updated

REDACTED

REDACTED

Sincerely yours,

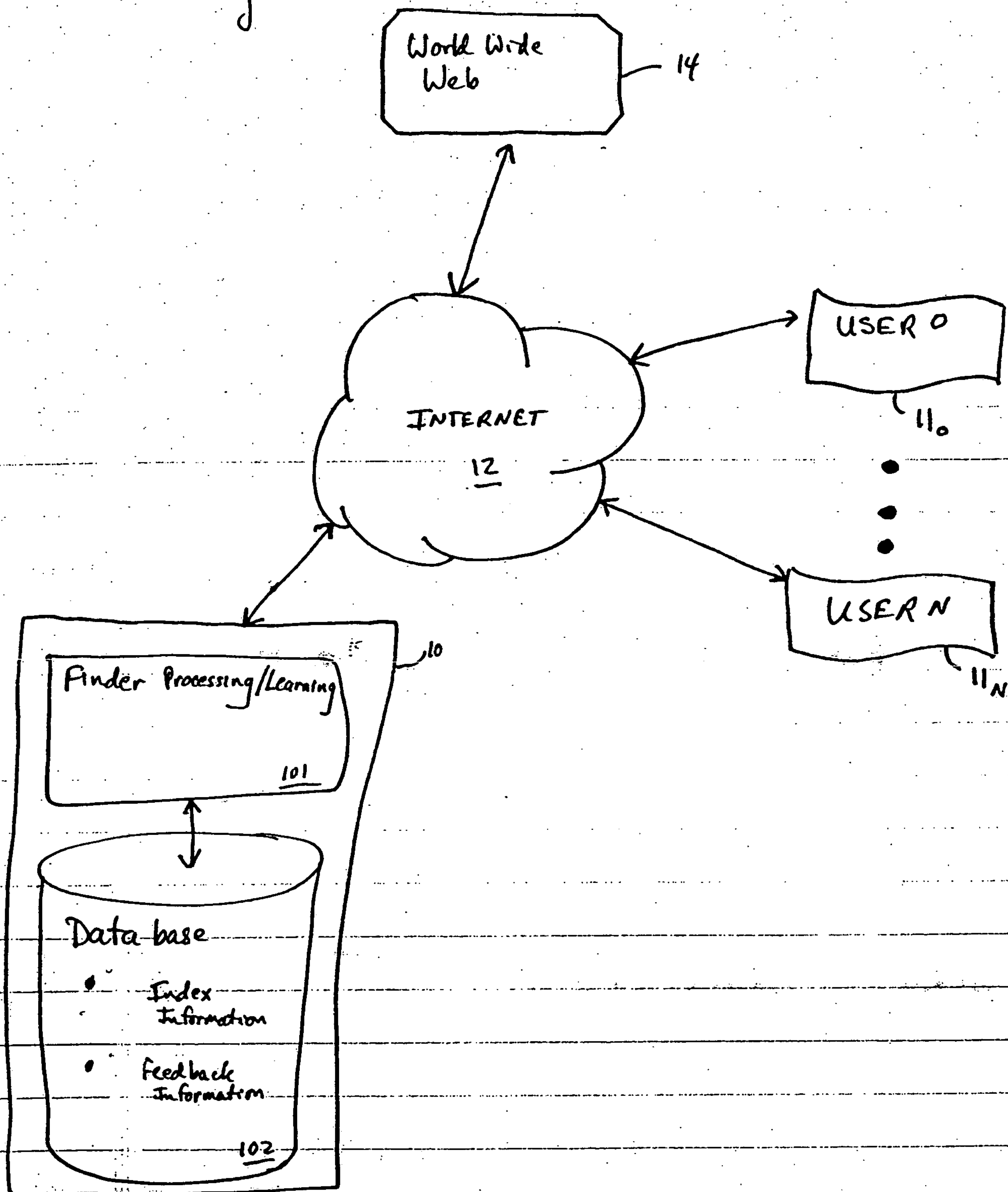
Joe

Joseph W. Ragusa

cc: Bruce G. Bernstein, Esq.

NY_MAIN 59967 v 1

Fig. 1A


World Wide Web
14
INTERNET
12
USER 0
11_0
USER N
11_N
10
Finder Processing/Learning
101
Data-base
• Index Information
• Feedback Information
102

TITLE

METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is directed to a computer-implemented product for locating and connecting to a particular desired object or target resource from among plural resources resident at distributed locations on a network.

Description of the Related Art

The worldwide network of computers known as the Internet evolved from military and educational networks developed in the late 1960's. Public interest in the Internet has increased of late due to the development of the World Wide Web (hereinafter, the Web), a subset of the Internet that includes all connected servers offering access to hypertext transfer protocol (HTTP) space. To navigate the Web, browsers have been

developed that give a user the ability to download files from Web pages, data files on server electronic systems, written in HyperText Mark-Up Language (HTML). Web pages may be located on the Web by means of their electronic addresses, known as Uniform Resource Locators (URLs).

A URL uniquely identifies the location of a resource (web page) within the Web. Each URL consists of a string of characters defining the type of protocol needed to access the resource (e.g., HTTP), a network domain identifier, identification of the particular computer on which the resource is located, and directory path information within the computer's file structure. The domain name is assigned by Network Solutions Registration Services after completion of a registration process.

While the amount of information available on the Web is enormous, and therefore potentially of great value, the sheer size of the Web makes the search for information, and particular web sites or pages, a daunting task. Search engines have been developed to assist persons using the Web in searching for web pages that may contain useful information.

Search engines fall into two major categories. In search engines falling into the first category, a service provider compiles a directory of Web sites that the provider's editors believe would be of interest to users of the service. The Yahoo site is the best known example of such a provider. Products in this category

are not, strictly speaking, search engines, but directories, and will be referred to hereinafter as “editor-controlled directories”. In an editor-controlled directory, the developer of the directory (the “editor”) determines, based upon what it believes users want, what search terms map to what web pages.

The other major category, exemplified by Altavista, Lycos, and Hotbot, uses search programs, called "web crawlers", "web spiders", or "robots", to actively search the Web for pages to be indexed, which are then retrieved and scanned to build indexes. Most commonly, this is done by processing the full text of the page and extracting words, phrases, and related descriptors (word adjacencies, frequencies, etc.). This is often supplemented by examining descriptive information about the Web document contained in a tag or tags in the header of a page. Such tags are known as “metatags” and the descriptive information contained therein as “metadata”. These products will be referred to hereinafter as “author-controlled search engines,” since the authors of the Web documents themselves control, to some extent, whether or not a search will find their document, based upon the metadata that the author includes in the document.

Each type of product has its disadvantages. Author-controlled search engines tend to produce search results of enormous size. However, they have not been reliable in reducing the large body of information to a manageable set of relevant results. Further, web site authors often attempt to skew their site's position in

the search results of author-controlled search engines by loading their web site metatags with multiple occurrences of certain words commonly used in searches.

Editor-controlled directories are more selective in this regard. However, because conventional editor-controlled directories do not actively search the web for matches to particular search terms, they may miss highly relevant web sites that were not deemed by the editors to be worthy of inclusion in the directory. Also, it is possible for the editor to "play favorites" among the multitude of Web documents by mapping certain Web documents to more search terms than others.

Recently, search engines such as DirectHit (www.directhit.com) have introduced feedback and learning techniques to increase the relevancy of search results. DirectHit purports to use feedback to iteratively modify search result rankings based on which search result links are actually accessed by users. Another factor purportedly used in the DirectHit service in weighting the results is the amount of time the user spends at the linked site. The theory behind such techniques is that, in general, the more people that link on a search result, and the longer the amount of time they spend there, the greater the likelihood that users have found this particular site relevant to the entered search terms. Accordingly, such popular sites are weighted and appear higher in subsequent result lists for the same search terms. The Lycos search engine (www.lycos.com) also uses feedback, but only at the time of crawling, not in

ranking of results. In the Lycos search engine, as described in U.S. Patent No. 5,748,954, priority of crawling is set based upon how many times a listed web site is linked to from other web sites. This idea of using information on links to a page was later exploited by the Clever system developed in research by IBM, and the Google system (www.google.com), which do use such information to rank possible hits for a search query.

Even leaving aside the drawbacks discussed above, search engines of both categories are most useful when a user desires a list of relevant web sites for particular search terms. Often, users wish to locate a particular web site but do not know the exact URL of the desired web site. Conventional search engines are not the most efficient tools for doing this.

Moreover, naming and locating particular sites on the Web is currently subject to serious problems. For example, appropriate names, including existing company names or trademarks, may not be available, because someone registered them first. Names may be awkward and not obvious, because of length, form/coding difficulties or variant forms, and names may not justify a separate domain name registration for reasons of cost and convenience, such as movie titles or individual products.

This problem results from a mismatch between the present network addressing scheme based on Uniform Resource Locators (URLs), which meet the technical

needs of the Internet software, and the needs of human users and site sponsors for simple, user-friendly mnemonic and branded names. This problem is largely hidden in cases where a user finds a site by clicking a pre-coded link (such as after using a search engine), or by using a saved bookmark. However, the problem does seriously affect users wishing to find a site directly, or to tell another person how to find it. To do this, the person must know and type the URL into his Internet browser, typically of the form sitename.com or www.sitename.com. Site sponsors are also seriously hampered by this difficulty in publicizing their sites.

Further, the current method of naming and locating Web sites has serious, widely known problems. Web site locator "domain" names are often not simple or easily remembered or guessed, and often do not correspond to company, trademark, brand or other well-known names.

As a result of the foregoing, site URLs (or domain names) are not intuitively obvious in most cases, and incorrect access attempts waste time and produce cryptic error messages that provide no clue as to what the correct URL might be. A significant percentage of searches are for specific, well-known sites. These could be found much more quickly by a special-purpose locator engine. The current mode of interacting with search engines is also cumbersome-for this purpose, a much simplified mode of direct entry is practical.

One attempt to provide the ability to map a signifier, or alias, to a specific URL utilizes registration of

key words, or aliases, which when entered at a specified search engine, will associate the entered key word with the URL of the registered site. One such commercial implementation of this technique is known as NetWord (www.netword.com). However, the NetWord aliases are assigned on a registration basis, that is, owners of web sites pay NetWord a registration fee to be mapped to by a particular key word. As a result, the URL returned by NetWord may have little or no relation to what a user actually would be looking for. Another key word system, RealNames (www.realnames.com), similarly allows web site owners to register, for a fee, one or more "RealNames" that can be typed into browser incorporating RealNames' software, in lieu of a URL. Since RealNames also is registration based, there is no guarantee that the URL to which is user is directed will be the one he intended.

Further, in existing preference learning and rating mechanisms, such as collaborative filtering (CF) and relevance feedback (RF), the objective is to evaluate and rank the appeal of the best *n* out of *m* sites or pages or documents, where none of the *n* options are necessarily known to the user in advance, and no specific one is presumed to be intended. It is a matter of interest in any suitable hit, not intent for a specific target. Results may be evaluated in terms of precision (whether "poor" matches are included) and recall (whether "good" matches are not included).

A search for "IBM" may be for the IBM Web site, but it could just as likely be for articles about IBM as a

company, or articles with information on IBM-compatible PCs, etc. Typical searches are for information *about* the search term, and can be satisfied by any number of "*relevant*" items, any or all of which may be previously unknown to the searcher. In this sense there is no *specific* target object (page, document, record, etc.), only some open ended set of objects which may be useful with regard to the search term. The discovery search term does not signify a single intended object, but specifies a *term* (which is an attribute associated with one or more objects) presumed to lead to any number of relevant items. Expert searchers may use searches that specify the subject indirectly, to avoid spurious hits that happen to contain a more direct term. For example, searching for information about the book Gone With The Wind may be better done by searching for Margaret Mitchell, because the title will return too many irrelevant hits that are not about the book itself (but may be desired for some other task).

In other words, the general case of discovery searching that typical search engines are tuned to serve is one where a search is desired to return some number, n, of objects, all of which are relevant. A key performance metric, *recall*, is the completeness of the set of results returned. The case of a signifier for an object, is the special case of n=1. Only one specific item is sought. Items that are not intended are not desired--their relevance is zero, no matter how good or interesting they may be in another context. The top DirectHit for "Clinton" was a Monica Lewinsky page. That is probably not because people searching for

Clinton actually intended to get that page, but because of serendipity and temptation—which is a distraction, if what we want is to find the White House Web site.

In addition,

-CF obtains feedback from a group of users in order to serve each given user on an overall, non-contingent basis--without regard to the either the intent of the user at a specific time, or to being requested in a specific context.

-RF is used by a single user to provide feedback on their intent at a given time, but still with no presumed intent of a single target.

More broadly, searching techniques are generally not optimized based on using a descriptor which is also an *identifier*--they provide more generally for the descriptor to specify the nature of the content of the target, not its name. There are options in advanced search techniques which allow specification that the descriptor is actually an identifier, such as for searching by title. Such options may be used to constrain the search when a specific target happens to be intended, but no special provision is made to apply feedback to exploit that particular relationship or its *singularity*.

Moreover, none of the currently available key word systems utilize heuristic techniques actually to determine the site intended by the user. Instead, the current systems teach away from such an approach by their use of registration, rather than user intention,

to assign key words to map to web pages. Thus, the current techniques are not directed to solving the problem of finding the one, correct site for a particular signifier.

Thus, the need exists for a system that would enable a user to find a desired Web document by simply entering an intuitive key word or alias and that would perform a one to one mapping of the alias with the URL actually desired by the user, and which would use heuristic techniques to assist in providing the correct mapping, and improving system accuracy over time.

SUMMARY OF THE INVENTION

In consideration of the above deficiencies of the prior art, it is an object of the present invention to provide a method of signifier mapping that allows a user to locate to a particular network resource, in the preferred embodiment a web page, by simply entering a signifier or alias.

Thus, the present invention is generally directed to a technique for intelligent searching or matching where a *signifier* is given and is to be related to a name or address of an *intended* target object.

Signifier, in the context of the present invention means:

-an identifier, referent, or synonym for the name or address of a specific resource (a target object) presumed to exist in some domain; but

-not necessarily a "name" or "address"--a canonical identifier that has been assigned by some authority or pre-set by some convention (names are a subset of signifiers--those which are canonical or pre-established);

-not necessarily a description of content or subject matter (concepts or words);

-an identifier that has cognitive significance to the user, and presumed communication value in identifying the intended target object to another person or intelligent agent.

In addition, this cognitive/communication value is based on a perceived relationship (meant to have minimal ambiguity) to an identifier, which might be an assigned name or a name based on common usage, but which need not be exact, as long at it serves to signify the intended target.

More generally, descriptors may possibly be considered to be signifiers, if they are intended to be unique or minimally ambiguous (e.g. "the company that commercialized Mosaic" or "the company that sells the ThinkPad").

It is a further object of the present invention to provide a system in which heuristic techniques are used together with user feedback to improve the accuracy of signifier mapping.

None of the many solutions to the signifier mapping problem (Netword, Centraal, Goto, etc.) have identified

learning as a valuable technique. This may be because what naturally come to mind are techniques based on pre-defined mappings that make the use of "de jure" explicit registration. That teaches away from the idea of trying to learn the mappings heuristically from colloquial usage. (The same applies to attempts at creating systems for "user friendly names" in other directory systems.) Since the mappings are understood as being defined or registered, why would one try to learn about them? But actually, the mappings are just like natural language—they are dynamic, evolving, and ambiguous, and can only be resolved in terms of *learned usage within a context* – which is best addressed by learning, as in the present invention, not registration or other static mappings as appear in the prior art.

The use of heuristic, adaptive feedback-based techniques operates in significantly different ways when focused on signifier mapping, and this can be exploited by isolating such tasks. The key difference between the present invention and most common searching tasks is that in the prior searching techniques, there is no intention of a specific target object that is known to exist.

The present invention has several advantageous features, various combinations of which are possible:

1) a special purpose mapping engine for locating popular sites by guessed names;
2) automatic display of the target site (if located with reasonable confidence);

3) an optional simplified mode of direct entry of a guessed site name; and

4) use of user expectations, such as popularity of guesses intended for a given site, as a primary criterion for translating names to sites, with provision for protection of registered trademarks or other mandates.

In accordance with one aspect of the present invention, a finder or locator server is established. The server is configured to work with a user interface that allows users to enter an guessed name or alias, as easily as if the user knew the correct URL for the intended target resource. In response to entry of the alias, the finder server accesses a database that includes, in a preferred embodiment, conventional Web-crawler-derived index information, domain name registration information, as well as user feedback from previous users of the server, and looks up the correct URL, i.e., the one URL that corresponds to the alias and causes the user's browser to go automatically to that URL, without the user having to view and click on a search results page, if the correct URL can be determined with a predetermined degree of confidence.

In one preferred embodiment, the server is structured to accept the alias as a search argument and do a lookup of the correct URL and the return of same to the browser, without the intermediate step of the user having to wait for and then click on a search results web page. The automatic transfer is preferably effected using standard HTML facilities, such as a

redirect page or framing. Redirect is effected by placing pre-set redirection pages at the guess URL on the server. Alternately, the redirect page can be generated dynamically by program logic on the server that composes the page when requested.

The present invention advantageously uses feedback and heuristic techniques to improve the accuracy of the determination of the correct URL. If a suggested match is found by the look-up technique and the accuracy of the mapping is confirmed by user feedback, then, after directing the user to the URL, the result is stored in the server to improve the accuracy of subsequent queries. The server database includes a list of expected terms and expected variants that can initially be catalogued to provide for exact matches. This list is updated by the learning processes discussed in more detail below.

If, on the other hand, a probable one intended match cannot be determined, the finder server preferably uses intelligent techniques to find a selection of links to possible matches ranked in order of likelihood, or could return a no-match page with advice, or a conventional search interface or further directories.

According to a preferred embodiment of the invention, each of the selection of links are configured not to go directly to the target URL. Rather, the links are directed back to a redirect server established by the finder server, with coding that specifies the true

target, and feedback information. The finder server can in this way keep track of user selections.

In accordance with an advantageous aspect of the invention, such feedback information is used to improve the results of the search by promoting web sites almost universally selected to exact match status, and by improving the ranking of possible lists in accordance with which links are most often selected. Preferably, a confidence parameter can be generated from such tracking to control whether to redirect to a URL or to present a possible list to users.

In furtherance of the above and other objects, there is provided, a designated server, accessible on the Internet, the designated server being configured to respond to relocation requests that specify an identifier, corresponding to a target resource, that may not be directly resolvable by standard Internet Protocol name resolution services to the URL of the target resource. In a direct entry embodiment of the present invention, requests are passed to the relocation server by sending a relocation URL that designates the relocation server as the destination node and appends the identifying information for the identifier as part of a URL string. The relocation server extracts the identifying information and translates it into a valid URL for the target resource. The relocation server is configured, in the event that a unique URL can be determined with respect to the target resource, to cause the target resource to be

presented to the user without further action on the part of the user.

Preferably, the user requests are entered at a web browser, the relocation or search server determines the valid URL for the target resource by performing a look-up in a database, and the response from the relocation server is in the form of a redirect page that causes the user's web browser to obtain the target resource.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A is an architectural block diagram of a server computer system internetworked through the Internet in accordance with a preferred embodiment of the present invention;

Figure 1B is a flow diagram illustrating a method of obtaining feedback from multiple users to be applied in searching or signifier mapping;

Figure 2 is flow diagram showing a method of signifier mapping using feedback and heuristics to continually improve the performance of the mapping;

Figure 3 shows an example of a database entry for the finder server of the present invention;

Figure 4A is a flow diagram illustrating a technique of feedback weighting for probable results in signifier mapping;

Figure 4B is a flow diagram illustrating a technique of feedback weighting for possible results in signifier mapping; and

Figure 5 is a flow diagram illustrating how feedback is used in a preferred embodiment to discriminate a probable target resource in accordance with the present invention.

DETAILED DESCRIPTION OF THE DISCLOSED EMBODIMENTS

"Population cybernetics" and the Internet

As a general matter, the present invention relates to a technique that collects experience (a knowledge base) from a mass population that is open ended or universal, either over all domains, or over some definable subject or interest domain or strata. This represents a significant improvement over prior art techniques, which are generally limited in the scope of the population and extent of experience from which they draw their knowledge base.

The technique of the present invention, in a preferred embodiment, uses the Internet to do this in a way that is powerful, economical, and far-reaching. The technique, in the preferred embodiment, uses the Internet to enable collection and maintenance of a far more complete knowledge base than has been used with any prior technique except Collaborative Filtering (CF).

In the present invention feedback learning is advantageously utilized, so that the information is not just collected, but refined based on feedback on the accuracy of prior inferences.

In its broad sense the present invention constitutes a kind of "population cybernetics," in that the learning does not just collect a linear knowledge base, but uses a feedback loop control process to amplify and converge it based on the results of prior inferences, and that it works over an entire population that is open, infinite, and inclusive. This is in contrast to prior learning techniques, which draw on necessarily finite, closed populations.

The use of population group information to achieve signifier mapping differs from the prior art technique of collaborative filtering in at least the following manner:

Whereas both CF and the technique of the present invention draw on knowledge of a population group to make inferences, CF obtains *ratings* of many things by many people to suggest other things (that may also be highly rated by the user, based on correlation with the group), and CF *does not involve* a *specific input request*, but rather seeks a *new, previously unknown item* in a category. On the other hand, the present invention obtains *translations* of many signifiers by many people to suggest the intended translation of a signifier and *involves a specific input request* to be translated to identify a *known intended target*

Although the technique of signifier mapping will occasionally be referred to loosely as searching, it is more accurately translation, because the target is intended and known, just not well specified. This differs from typical Web or document searching, which typically seeks unknown, new items.

The technique of the present invention also differs from natural language (NL) translation or understanding, in that the input is *atomic* in that it is no context as part of a body of discourse (a text). NL understanding techniques on the other hand translate words as components of concepts embedded in texts having a *context* of related ideas. Thus the cues of context in a discourse are absent, and the translation must be done without any such cues (although knowledge of the user may provide a useful context of behavior, demographics, psychographics that has some value in inferring intent). The task is to infer or predict intention, rather than to understand meaning, because there is no basis to infer meaning in any conceptual sense. The input is disjointed from any context, and if not seen before (from the user or others), there is little useful information on either its meaning or intention. The present invention seeks to infer intention based on limited data, primarily the input request, and draws on group data (of request translations) as its strength.

The task of the present invention has similarities with cryptanalysis, in that both the present invention and

cryptanalysis use data about communications behavior from groups of communicators to make inferences. However the task differs in that

- Cryptanalysis deals with intentional hiding of meaning or intention, where the technique of the present invention is applied to cases where the hiding (of the intention of a signifier) is not at all intended; and
- Cryptanalysis seeks to infer meaning (ideas) drawing on context in a discourse, like NL understanding, not usually to infer the intention of a signifier (of objects or actions) which is not in a context.

This point of intention versus meaning is subtle, but has to do with communication of commands or requests as opposed to concepts.

- One view of this is the idea of "requests," as opposed to declarations or assertions, in the use of language.
- This task of recognizing commands (vs. meanings) has parallels in the task of robot control, such as that based on spoken commands. The similarity is in training understanding of the speech of many users to speaker independent, and to infer meanings of a current speaker from that of others. The difference is that the tasks addressed in the present invention deal with a very wide, effectively infinite universe of commands (intended objects), while robot control techniques have generally been limited to very small sets of commands (partly because of the inability to apply mass experience).

Thus the technique of the present invention could be viewed as addressing a special class of robot control (in which experience data and feedback is accessible), and may ultimately be extensible to other robot control applications as such data becomes accessible over the network.

The social dimension is critical for inferences relating to shared objects or resources. Names draw on social conventions and shared usage. This social usage information is essential to effective mapping of signifiers to resources. De-jure naming systems can underlie a naming system, as for current Internet domain names, but de-facto usage is the essential observable source of information for fullest use. De-jure systems suffer from entropy, corruption and substitution, while de-facto usage is pragmatic and convergent to changing usage patterns.

This applies to a variety of name-able resources:

- Web domain names;
- Web sub-site names (to find sub-areas);
- People or business names;
- Department, agent, or service identifiers (to find contact points);
- Policy capability specifications (to find permissions, such as someone who can provide access to a given resource for a given purpose, such as confirming employment status or update-access to a report);

- Information sets or collections (to find reference tools that are known to exist, such as an IBM dictionary of acronyms, or an index of papers in ACM publications);
- Other robot control tasks, as social experience and feedback becomes accessible.

Social usage information can be combined with other sources of information in a heuristic fashion. For example, there could be a hierarchy that might be used in order, as available:

1. Personal defined usage information, such a defined personal nicknames;
2. Public de-jure defined mappings or directories;
3. Personal usage information (a person's own undefined nicknames, learned from my own usage/feedback);
4. Social de-facto usage information;

This is just one possible sequence, but shows how the usage data can take searching beyond what has been defined.

As discussed above, a preferred embodiment of the present invention relates to a method and apparatus for locating a desired target resource located and accessible on a network, in response to user entry of a guessed name or alias. In illustrating the preferred embodiment, the apparatus is shown as a server computer, or computers, located as a node on the Internet. However, the present invention is in no way limited to use on the Internet and will be useful on

any network having addressable resources. Even more broadly, the present invention is useful for any similar task of identifying an intended target for an action in which automatic facilitation of that action is desired, where feedback from a large population can be obtained to learn whether a given response was in fact the one that was desired. Control of robots, as discussed herein, is one example of such broader application.

The finder server of the preferred embodiment of the present invention allows users to enter a guessed identifier or alias, as easily as if they knew the correct URL. Specifically, the finder server of the present invention accepts a guessed name, or alias, from a user, uses a look-up technique, enhanced by heuristics preferably taking into account previous users' actions, to determine a correct URL for the intended target resource, and causes the user's browser to go to that URL automatically. Preferably this is done without the added step of first viewing and clicking on a search-results page, where an initial search finds the intended target resource with a predetermined degree of certainty. Such a resource will be referred to hereinafter as a "probable". In accordance with a preferred embodiment of the present invention, this functionality can be implemented by:

- Publicizing the locator server under an appropriate URL name, for example, guessfinder.com.
- Setting up the server to, in response to entry of a guessed name or alias, do a lookup to the correct URL and return a response that causes the user's browser

to go automatically to the specified URL. Such an automatic transfer can be effected using a standard HTML facilities, such as a redirect page, or framing.

- If the guess does not provide an exact match in the lookup phase, using feedback and heuristic techniques to create and present to the user a selection of links to possible matches. Alternately, the user may be presented with a nomatch page with advice, or directed to a conventional search interface, or further directories.

It is contemplated that the use of aliases for attempting to locate a web site associated with company name or brand name would be found useful. For example, the aliases "s&p", "s-p", "sandp", "snp", "standardandpoors", "standardnpoors", "standardpoors" should preferably all map to www.standardpoors.com. In addition to companies and brands, other important name domains would include publications, music groups, sports teams, and TV shows.

The present invention advantageously provides for learning and feedback on the basis of user preferences to automatically and dynamically build a directory of names and sites that maps to the actual expectations and intentions of a large population of users, and adapts to changes over time, including the appearance of new sites, thus optimizing utility to them.

The finder server of the present invention effectively provides a secondary name space, administered by the organization operating the finder system, through the

automated heuristic methods described here, that maps to, but is not dependent on, the URL name space. The finder site computer has access to a data base containing entries for any number of popular sites, with any number of likely guesses and variations for each site.

As a result of the service provided when the present invention is implemented, site sponsors could skip the cumbersome and costly process of obtaining specific mnemonic URLs or alternate URLs in many cases (especially with regard to domain names). Even with a number of conventional URLS, this service could be a supplement, for additional variations. The problem of pre-empted URL domain names would also be avoided, except where there is legitimate and significant pre-existing usage.

A key to utility is to be able to directly connect in response to most guesses, and ambiguities could be a limiting factor. To avoid that it is desirable to exploit Pareto's Law/the 80-20 rule and do a direct connect even when there is an uncertain but likely target. For that to be useful, it must be easy for users to deal with false positives.

Correction after arrival at a wrong site can be made relatively painless by allowing a subsequent request to indicate an error in a way that ties to the prior request and adds information. For example a request, guessfinder.com/lionking, that located the movie but was meant to find the play could be corrected by

entering guessfinder.com/lionking/play. A more efficient coding might explicitly indicate an error, such as guessfinder.com/!/lionking/play. Even with the error, this would be quicker and easier than conventional methods.

Correction in-flight can be achieved by using the existing visibility of the redirect page, or enhancing it. When a redirect page is received by a user's browser, it appears for a short time (as specified with an HTML refresh parameter) while the target page is being obtained. In addition to advertising content, that page preferably lists the redirection target, as well as alternatives, allowing the user to see the resolution in time to interrupt it. This is most useful with a browser that permits a redirect to be stopped in mid-stream by clicking the stop button, leaving the redirect page on display, and allowing a correct selection among alternative links to be made.

Note some of the typical parameters and control points that would be relevant:

"New" sites.

Applies when the user wants a site but is provided neither a direct hit, nor a correct possible. Users would find the site via alternate means (offered through the service or not). The user then submits an add-site request, via the Web or e-mail. If the number of add-site requests over a set interval exceeds a set (low) threshold, the site is added as a possible, or a direct if there are no competing alternatives. Such

adds would be provisional, and could be dropped if requests are not sustained.

Possibles

Low confidence possibles would be listed low on the list, and selections would be tracked. If selections are strong, they move up the list. If selections are very weak, they would drop off after some interval. The threshold to add back sites that were dropped might be higher for a time, to limit oscillation and false adds. If possibles are well ahead of alternatives by some threshold over some interval, they would be promoted to direct hits.

Direct hits

Feedback on false positives would be collected. This could be via links in frames, redirect pages, interstitials, or other means, as suggested previously. If false positives exceed a threshold, the site would revert to a possible and the common alternatives would be listed as well.

Parameter issues: thresholds, intervals, smoothing, damping, overrides.

Basic parameters include the various thresholds and time intervals for measurement. Smoothing techniques (such as exponential smoothing) would be applied to adjust for random variations and spikes, to improve forecasting. Damping mechanisms could be used to limit undue oscillation from state to state. Overrides would provide for mandated or priority matches, such as for

registered trademarks, on either a weighted or absolute basis, as appropriate.

Figure 1A illustrates a first embodiment of the present invention, as implemented on the Internet. The finder server 10 includes a computer or computers that perform processing, communication, and data storage to implement the finder service. Finder server 10 includes a finder processing/learning module 101. Module 101 performs various processing functions, and includes a communication interface to transmit and receive to and from the Internet 12, as well as with database 102, and is programmed to be operable to learn from experiential feedback data by executing heuristic algorithms. Database 102 stores, in a preferred embodiment, indexes of URL data that would allow the module 101 to locate, with a high degree of confidence, a URL on the Web that is an exact match for a target resource in response to a user's entry of an alias or guessed name. Preferably, the indexes store, in addition to available URL information, such as domain name directories, information relating to the experience of the server in previous executions of the finder service. As the server gains experience and user feedback, heuristic techniques are applied by module 101 to enable the returned URLs to conform more and more accurately to user expectations.

Users 11_0-11_N can access the Internet 12 by means of client computers (not shown) either directly or through an Internet service provider (ISP).. As has been discussed previously, to make use of the present

invention the user enters a guessed name, or alias, into his computer's browser and submits a query containing the alias to the finder server. The World Wide Web 14 includes computers supporting HTTP protocol connected to the Internet, each computer having associated therewith one or more URLs, each of which forming the address of a target resource. Other Internet information sources, including FTP, Gopher and other static information sources are not shown in the figure.

The finder server includes operating system servers for external communications with the Internet and with resources accessible over the Internet. Although the present invention is particularly useful in mapping to Internet resources, as was discussed above, the method and apparatus of the present invention can be utilized with any network having distributed resources.

Entry of the alias by a user may be accomplished in a number of ways. In one embodiment, a usage convention can be publicized for passing the alias to the server within a URL string, such as guessfinder.com/get?ibm, for example, for trying to find the web page corresponding to the alias "ibm". In this case, the server is programmed to treat the string "ibm" as a search argument and perform the appropriate processing to map the alias to the intended target resource. A similar effect can be obtained by the somewhat simpler form guessfinder.com/ibm, if the server is programmed appropriately. Alternately, the user can visit the web site of the finder server and be presented with a

search form, as is done in conventional search engines. A third option is to provide a browser plug in that allows direct entry of the key word in the browser's URL window or any alternative local user interface control that will then pass the entry on as a suitably formatted HTTP request.

It also would be preferable for an enhanced user interface to be phased in as the service gains popularity. This preferably would be accomplished by a browser plug-in, or modifications to the browser itself, to allow the alias to be typed into the URL entry box without need for the service domain name prefix (such as, guessfinder.com/...). Instead, such an entry would be recognized as a alias, not a URL, and the prefix would be appended automatically, just as http://... is appended if not entered with a URL in current browsers.

Figure 1B is a flow diagram illustrating a technique for obtaining and learning from feedback responses gathered from a large group of people, in the example, users 1, 2, . . . n. Such a technique can be used in a variety of applications, and in particular in traditional search engines, or in mapping to identify particular web sites, as in alias or signifier mapping.

In Figure 1B, users 1, 2, . . . n represent a large community of users. In the flow diagram, the flow of query items from the users is indicated by a Q, the flow of responses back to the users is indicated by an R, and the flow of feedback results provided by the

users' actions, or responses to inquiries, is indicated by an F. As can be seen from the figure, Query (a, 1) is transmitted from user 1 to the service 2, which can either be a searching or a mapping service. The service has learning processor 4, which interfaces with a database 6. The database 6 contains, among other things, indexes and feedback information gathered from previous queries. In response to the query, the user 1 is provided with a response R(a, 1). User 1 then is provided with the opportunity to transmit user Feedback (a, 1) to the Service 2. Learning processor 4 stores the feedback information in the database 6, and is programmed with one or more heuristic algorithms enabling it to learn from the feedback information to improve the returned search or mapping results. The feedback provided will improve the results offered, for example by positively weighting results preferred by users, so that, over time, more accurate results can be obtained.

Figure 2 is a diagram illustrating the logical flow used in applying the general technique of learning from user feedback shown in Figure 1 to signifier mapping, in accordance with a preferred embodiment of the present invention. A user enters a Query consisting of a signifier, represented by Q^s. The server, in response to receipt of the query, parses the query, at step S02, and in step S04 performs a database lookup in an attempt to determine, if possible, the exact target resource intended by the user. Database 6 includes index data as well as feedback data obtained from users in previous iterations of the signifier mapping

program, is accessed. The stored data structure is described in more detail below.

In step S06, the program discriminates a probable intended target making use of the index information such as domain registration indexes, and other resources, as well as the feedback information stored in the database. In step S08, if a likely hit, or exact match has been identified, that is, a web page has been located with a high confidence parameter, the flow continues to step S10. At step S10, a direction is prepared to the likely hit URL. A list of alternatives optionally may be provided for presentation to the user at the same time, in case the likely hit turns out not to be the target identifier. At step S12, the server sends information R^s to the user, more particularly to the user's browser, to effect a link to the likely hit. Optionally, the alternate list is also provided at the same.

In step S14, the viewed page is monitored by the server and the user, by his actions, provides feedback. Most readily determined with no assistance from the user is the fact of the user having chosen the link. This may be determined, for example by a redirect, in which an intermediate server is transparently interposed between the browser and the target page, and thus able to identify the user and the URL target based on coding built into the URL that the user clicks. Also desirable is the amount of time the user spends at the site, which will be an indicator of whether the site is the intended target. This may be ascertained, for

example, if clickstream data can be obtained, such as through the use of a monitor program that works as a browser add-in or Web accessory, such as the techniques offered by Alexa. Other feedback can be provided by asking the user. This can, for example, be done conveniently by using a small header frame served by the relocation service that appears above the actual target page, and that includes controls for the user to indicated whether or not the results were correct. The URL of the viewed page is recorded, together with any other feedback, for use in improving the accuracy of subsequent iterations of signifier mapping. At step S26, the feedback data is supplied to a feedback weighting algorithm, described in detail below, which generates appropriate weighting factors to be stored in the database for use in subsequent mappings.

If it is determined at step S08 that the result is not a likely hit, the flow proceeds to step S18, where a list of the top m hits (m being a predetermined cutoff number), preferably drawing on the list of possible hits from a conventional search engine, or by employing the same techniques as a conventional search engine, is prepared. Unlike conventional search engines, the ranking of these hits is based primarily on experience feedback data as described below. In addition, where such feedback is limited or absent, it would be supplemented by variants of more conventional search engine weighting rules that expressly tuned to the task of finding a single intended result (i.e., high relevance by low recall) rather than many results (high relevance plus high recall). The list is presented, at

step S20, to the user as R^s. The user, by the selections made from the provided list, and from other feedback, such as how long the user spends at each link, supplies feedback to the system. This information F^s is monitored, at step S22 and recorded, at step S24. The recorded information is supplied to the feedback weighting algorithm, at step S26, the output of which is stored in the database for use in subsequent iterations of the signifier mapping.

Figure 2 illustrates the simple case in which a user is directed to a target URL if the target has been determined to be a probable hit, and is presented with a list to choose from if the target cannot be identified with sufficient certainty. However, it is well within the intended scope of the invention for alternate methods to be employed. For example, the user interface (UI) could be extended, either by framing, or a browser plug-in or extension, to provide multi-pane/multi-window results that allow a pane for each type of response, e.g., the target response and a list of possibles, regardless of the level of confidence in the result. In such a case, the format for presentation of results would be the same whether a probable has been located or not, but the learning from feedback and ranking would still seek to determines "correctness" based on the varying feedback cases.

Figure 3 illustrates a preferred method of organizing index data to allow for storing and updating of the most probable hits for a given query. As can be seen from the illustration, for each query, whether single

element queries or compound queries, there is stored a list of associated possible targets. Linked to each of these query/target pairs is a raw score, an experience level, and a probability factor. As feedback enters the system, the index data is updated to reflect the user feedback. The updating process will be described below. While the index shows preferred weighting criteria, these are only a sample of the kind of criteria that can be correlated to the query/target pairs. In a simple embodiment, the raw score would be based only on selections of hits, and explicit feedback and correctness as described below. Other embodiments could add feedback data on time spent at a target. Additional variations would include weighting based on the recency of the feedback, and on the inclusion of non-feedback data, such as the various syntactic and semantic criteria used for relevance weighting by conventional search engines.

The process of maintaining the guess-target database is adaptable to a high degree of automation, and this can be highly responsive to new sites. An outline of such a method is:

All guesses are logged and analyzed.

Ambiguous hits are tracked as described earlier.

Complete non-matches are sorted by frequency to identify common new requests (in real time). Changes in ambiguous match patterns could also flag appearance of new sites.

Common new requests preferably are fed to an automated search tool that would use existing search

engines, hot site lists, and name registration servers, etc. to identify possible targets.

Automated intelligent analysis of those results can seek to qualify probable targets.

High confidence (or possible) targets preferably are added, and then tracked based on the feedback mechanism described earlier, in order to self-correct. A confidence parameter preferably is used to control whether to redirect or to present a possibles list to users.

Human review and correction also preferably is used to supplement this.

Figure 4A illustrates a preferred technique for weighting the results using feedback data for hits that have been determined to be probable hits. In step S30, if the user feedback from the probable result indicates that the probable was in fact the target URL the user was searching for, the flow proceeds to step S32 where the raw score for that query/target pair is incremented by factor_Y. If the user returns feedback indicating that the probable was not the target resource the user had in mind, the flow proceeds to step S34 where the raw score for that query/target pair is decremented by factor_N. If the user provides no feedback, then the flow proceeds to step S36 where the raw score is decremented by factor_0, which can be zero. After execution of any of steps S32, S34 or S36, the flow proceeds to step S38, at which the experience level score is incremented by Efactor_C.

Figure 4B illustrates a preferred technique for weighting in accordance with user feedback in the case of possibles, i.e., items on the list presented to the user when no probable result can be located. As shown in the figure, if a possible is selected by the user from the presented list, at step S40, the fact of selection is recognized, preferably by use of a redirect server that allows the system to keep track of which link was chosen. Additionally, the amount of time the user spends at the selected link may be ascertained. Making use of the information gathered in the redirect and such other feedback as may be obtained, the raw score for the query/target pair is incremented, at step S44, by factor_S. The user is then requested to provide additional feedback after the user has finished viewing the link.

In a preferred embodiment of the present invention, this feedback is gathered from the user by presenting the user with a frame that includes a mechanism, such as a check box, or radio button, that allows the user to indicate whether the selected possible was in fact the intended or "correct" target resource. If it is determined, at step S42, from the feedback that the link was the correct target, the flow proceeds to step S46, where the raw score for that query/target pair is incremented by $\text{factor}_{Y'}$. If the user returns a negative response, the raw score of the pair is decremented at step S48 by a by $\text{factor}_{N'}$. If no feedback is received, the raw score is decremented, at step S50, by $\text{factor}_{O'}$, which can be zero. After execution of any of steps S44, S46, S48 or S50, the flow proceeds to step S32, at

which the experience level score is incremented by $Efactor_{PS}$ in the case of selection of the link, and by $Efactor_{PC}$ if the link was the correct.

Figure 5 illustrates a detail of how the present invention ranks and discriminates a probable target. At step S100 a list of possibles is obtained. Next, the list is ranked, at step S102, on the basis of the expected probability as the target. In step S102, a discrimination criteria is calculated and compared with a predetermined threshold parameter. For example, if ProbTi is the expected probability that Ti is the correct target, a formula such as the example shown can be used to determine whether T1 stands out as more probable than T2 by a relative margin that exceeds a set threshold needed to judge it as the probable intended one target. When the threshold is not exceeded, the implication is that one of the secondary possibilities may very well be the intended one, and that directing the user to the slightly favored target may not be desirable.

In the preferred embodiment, when a link on a list of possibles is selected by the user, rather than connect the user immediately to the chosen link, the finder server first redirects the user to a redirect server where feedback data relating to the selection can be gathered. One item of feedback that may be obtained in this manner is the very fact of the selection. Further feedback can be obtained by additional means, such as monitoring how long the user spends at the selected link, and by directly querying the user.

The redirect linking technique uses the target URL as a server parameter within a composite URL to control the intermediate server parameter within the URL to control the intermediate server. The target URL is embedded as a server parameter within a URL that addresses the redirect server, and the URL parameter is used to control the intermediate server process. Thus a server is called with a first URL, a redirect URL, that specifies the second URL, i.e., the target URL, as a parameter. For example

http://redirector.com/redirector?query12345678/targetserver.com/targetpath1/targetpage1.htm

where redirector.com is the intermediate server URL, query12345678 is a unique identifier of the user-query combination, and targetserver.com/targetpath1/targetpage1.htm is the target URL. The network ignores the parameter portion of the URL, which is passed as data to the server. The server acts on the parameter to perform desired intermediary processing, in this case, the logging of the fact that this link was clicked in response to query12345678, and to redirect the user to the intended location specified by the second URL. The token query12345678 could be a unique identifier corresponding to a logged user-query entry, or it could be the actual query string.

The delay required for the redirect provides the opportunity for the display of interstitial advertisements. In addition, additional user feedback

can be solicited during the delay, and the connection to the targeted URL can be aborted if the user indicates that the target site is not the one he or she intended. In addition to using the redirect when a link is selected, the technique also preferably is used when an exact match is found, to provide a brief delay before connecting the user to the exact match, to present advertisements to give the user the time to abort the connection. In any event, the user preferably is given the opportunity to provide feedback after connecting to any site, whether directly as a result of an exact match, or as a result of selecting from a linked possibles list.

The redirect server of the present invention allows data to be gathered on each link as it is followed and redirected. The redirect link can be created in a simple static HTML. However, it is preferable to create the link dynamically for each user selection.

The finder is setup to recognize the feedback function, possibly as a CGI or other gateway/API function, and invoke the appropriate function to parse the URL or other data (referer, cookies, etc.), extract the target URL and feedback information for processing, and return a page containing a redirect (or use framing or other means) to take the user to the desired target.

This mechanism is general, and can be used for many purposes. In the case of the finder server:

-Reasonably complex feedback information can be obtained, which at minimum would include the original guess. Thus a log of each guess that was not clearly resolved, paired with the corresponding user-selected target, can be obtained.

-That set of selected guess/target pairs can then be used to adjust the confidence levels in the guess/target database. Similar data on directly resolved pairs would also be applied, along with any data from wrong-match reports.

Other applications are to any situation where links go to sites other than the source. This would include results of conventional search engines, as well as resource directories, sites referring users to suppliers, advertisers, etc.

The above embodiments of the present invention have been described for purposes of illustrating how the invention may be made and used. The examples are relatively simple illustrations of the general nature of the many possible algorithms for applying feedback data that are possible. However, it should be understood that the present invention is not limited to the illustrated embodiments and that other variations and modifications of the invention and its various aspects will become apparent, after having read this disclosure, to those skilled in the art, all such variations and modifications being contemplated as falling within the scope of the invention, which is defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method of finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained on the database, the method comprising:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

2. A method according to Claim 1, wherein a resource is determined at the accessing step to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

3. A method according to Claim 2, wherein if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined

confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

4. A method according to Claim 3, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

5. A method according to Claim 1, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

6. An apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database;

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

7. An apparatus according to Claim 6, wherein a resource is determined to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

8. An apparatus according to Claim 7, wherein the apparatus is operable to, if the guessed name does not map uniquely to any of the indexed resources, to a predetermined confidence level, perform the following steps:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

request that the user select a link; and

cause the user's computer to go to the selected link, if any.

9. An apparatus according to Claim 8, wherein the apparatus is operable to, if a link has been selected, perform the following steps:

add information regarding the selection of the link to the feedback information in the database;

solicit user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, increment the

confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

10. An apparatus according to Claim 6, the apparatus being further operable to:

solicit user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, increment the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

11. A system for finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising

a plurality of interconnected computers, the system comprising:

finder server means having: (a) database means for storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system; and (b) learning system means for accessing and learning from information contained on the database;

receiving means for receiving a guessed resource name from the user;

accessing means for accessing the database means to determine which, if any, of the indexed resources is the desired target resource;

directing means for directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined in the access means to be the target resource.

12. A system according to Claim 11, wherein a resource is determined by the access means to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

13. A system according to Claim 12, further comprising:

presenting means for, if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence

factor, the resources having the highest confidence factors being ranked highest;

requesting means for requesting that the user select a link from the presented list; and

means for causing the user's computer to go to the selected link, if any, upon a user selection of a link.

14. A system according to Claim 13, further comprising:

adding means for, if a link has been selected, adding information regarding the selection of the link to the feedback information in the database;

soliciting means for soliciting user feedback with regard to the selected link; and

means for, if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

15. A system according to Claim 11, further comprising:

soliciting means for soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

means for, if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the

confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

16. A computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method comprising:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

17. A computer-readable medium according to Claim 16, wherein a resource is determined in the accessing step to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

18. A computer-readable medium according to Claim 17, wherein if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

19. A computer-readable medium according to Claim 18, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not

the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

20. A computer-readable medium according to Claim 16, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

21. A system for finding resources on a network of interconnected computers on which a plurality of resources reside, the system comprising:

a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and

a finder server having:

(a) a database including: (i) an index of resources available on the network; and (ii)

information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and

(b) a learning system operable to access and learn from information contained on the database,

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

ABSTRACT OF THE DISCLOSURE

An apparatus for finding resources on a network comprises a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database. The finder server is operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by: receiving a guessed resource name from the user; accessing the database to determine which, if any, of the indexed resources is the desired target resource; directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

Date: 3/5/00
Sender: rreisman@teleshuttle.com (Richard Reisman)
To: Joseph W Ragusa, bgb@btgusa.com
Priority: Normal
Subject: Notes on CUL draft 2

Here are my notes.

REDACTED

-R

Richard Reisman - Teleshuttle Corporation
20 E 9 St.-14K, NY, NY 10003 - (212)-673-0225 (phone/fax) (Daytime: 646-227-0020 x122)
Web: http://www.teleshuttle.com e-mail: rreisman@teleshuttle.com
==

RR-CUL-Notes2.zip

PROPRIETARY AND CONFIDENTIAL

notes marked by {*} 2/26-3/4/00**

TITLE

METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is directed to a computer-implemented product for locating and connecting to a particular desired object or target resource from among plural resources resident at distributed locations on a network.

Description of the Related Art

The worldwide network of computers known as the Internet evolved from military and educational networks developed in the late 1960's. Public interest in the Internet has increased of late due to the development of the World Wide Web (hereinafter, the Web), a subset of the Internet that includes all connected servers

offering access to hypertext transfer protocol (HTTP) space. To navigate the Web, browsers have been developed that give a user the ability to download files from Web pages, data files on server electronic systems, written in HyperText Mark-Up Language (HTML). Web pages may be located on the Web by means of their electronic addresses, known as Uniform Resource Locators (URLs).

A URL uniquely identifies the location of a resource (web page) within the Web. Each URL consists of a string of characters defining the type of protocol needed to access the resource (e.g., HTTP), a network domain identifier, identification of the particular computer on which the resource is located, and directory path information within the computer's file structure. The domain name is assigned by Network Solutions Registration Services after completion of a registration process.

While the amount of information available on the Web is enormous, and therefore potentially of great value, the sheer size of the Web makes the search for information, and particular web sites or pages, a daunting task. Search engines have been developed to assist persons using the Web in searching for web pages that may contain useful information.

Search engines fall into two major categories. In search engines falling into the first category, a service provider compiles a directory of Web sites that

the provider's editors believe would be of interest to users of the service. The Yahoo site is the best known example of such a provider. Products in this category are not, strictly speaking, search engines, but directories, and will be referred to hereinafter as "editor-controlled directories". In an editor-controlled directory, the developer of the directory (the "editor") determines, based upon what it believes users want, what search terms map to what web pages.

The other major category, exemplified by Altavista, Lycos, and Hotbot, uses search programs, called "web crawlers", "web spiders", or "robots", to actively search the Web for pages to be indexed, which are then retrieved and scanned to build indexes. Most commonly, this is done by processing the full text of the page and extracting words, phrases, and related descriptors (word adjacencies, frequencies, etc.). This is often supplemented by examining descriptive information about the Web document contained in a tag or tags in the header of a page. Such tags are known as "metatags" and the descriptive information contained therein as "metadata". These products will be referred to hereinafter as "author-controlled search engines," since the authors of the Web documents themselves control, to some extent, whether or not a search will find their document, based upon the metadata that the author includes in the document.

Each type of product has its disadvantages. Author-controlled search engines tend to produce search

results of enormous size. However, they have not been reliable in reducing the large body of information to a manageable set of relevant results. Further, web site authors often attempt to skew their site's position in the search results of author-controlled search engines by loading their web site metatags with multiple occurrences of certain words commonly used in searches.

Editor-controlled directories are more selective in this regard. However, because conventional editor-controlled directories do not actively search the web for matches to particular search terms, they may miss highly relevant web sites that were not deemed by the editors to be worthy of inclusion in the directory. Also, it is possible for the editor to "play favorites" among the multitude of Web documents by mapping certain Web documents to more search terms than others.

Recently, search engines such as DirectHit (www.directhit.com) have introduced feedback and learning techniques to increase the relevancy of search results. DirectHit purports to use feedback to iteratively modify search result rankings based on which search result links are actually accessed by users. Another factor purportedly used in the DirectHit service in weighting the results is the amount of time the user spends at the linked site. The theory behind such techniques is that, in general, the more people that link on a search result, and the longer the amount of time they spend there, the greater the likelihood that users have found this particular

site relevant to the entered search terms. Accordingly, such popular sites are weighted and appear higher in subsequent result lists for the same search terms. The Lycos search engine (www.lycos.com) also uses feedback, but only at the time of crawling, not in ranking of results. In the Lycos search engine, as described in U.S. Patent No. 5,748,954, priority of crawling is set based upon how many times a listed web site is linked to from other web sites. This idea of using information on links to a page was later exploited by the Clever system developed in research by IBM, and the Google system (www.google.com), which do use such information to rank possible hits for a search query.

Even leaving aside the drawbacks discussed above, search engines of both categories are most useful when a user desires a list of relevant web sites for particular search terms. Often, users wish to locate a particular web site but do not know the exact URL of the desired web site. Conventional search engines are not the most efficient tools for doing this.

Moreover, naming and locating particular sites on the Web is currently subject to serious problems. For example, appropriate names, including existing company names or trademarks, may not be available, because someone registered them first. Names may be awkward and not obvious, because of length, form/coding difficulties or variant forms, and names may not justify a separate domain name registration for reasons

of cost and convenience, such as movie titles or individual products.

This problem results from a mismatch between the present network addressing scheme based on Uniform Resource Locators (URLs), which meet the technical needs of the Internet software, and the needs of human users and site sponsors for simple, user-friendly mnemonic and branded names. This problem is largely hidden in cases where a user finds a site by clicking a pre-coded link (such as after using a search engine), or by using a saved bookmark. However, the problem does seriously affect users wishing to find a site directly, or to tell another person how to find it. To do this, the person must know and type the URL into his Internet browser, typically of the form sitename.com or www.sitename.com. Site sponsors are also seriously hampered by this difficulty in publicizing their sites.

Further, the current method of naming and locating Web sites has serious, widely known problems. Web site locator "domain" names are often not simple or easily remembered or guessed, and often do not correspond to company, trademark, brand or other well-known names.

As a result of the foregoing, site URLs (or domain names) are not intuitively obvious in most cases, and incorrect access attempts waste time and produce cryptic error messages that provide no clue as to what the correct URL might be. A significant percentage of searches are for specific, well-known sites. These

could be found much more quickly by a special-purpose locator engine. The current mode of interacting with search engines is also cumbersome-for this purpose, a much simplified mode of direct entry is practical.

One attempt to provide the ability to map a signifier, or alias, to a specific URL utilizes registration of key words, or aliases, which when entered at a specified search engine, will associate the entered key word with the URL of the registered site. One such commercial implementation of this technique is known as NetWord (www.netword.com). However, the NetWord aliases are assigned on a registration basis, that is, owners of web sites pay NetWord a registration fee to be mapped to by a particular key word. As a result, the URL returned by NetWord may have little or no relation to what a user actually would be looking for. Another key word system, RealNames (www.realnames.com), similarly allows web site owners to register, for a fee, one or more "RealNames" that can be typed into browser incorporating RealNames' software, in lieu of a URL. Since RealNames also is registration based, there is no guarantee that the URL to which is user is directed will be the one he intended.

Further, in existing preference learning and rating mechanisms, such as collaborative filtering (CF) and relevance feedback (RF), the objective is to evaluate and rank the appeal of the best *n* out of *m* sites or pages or documents, where none of the *n* options are

necessarily known to the user in advance, and no specific one is presumed to be intended. It is a matter of interest in any suitable hit, not intent for a specific target. Results may be evaluated in terms of precision (whether "poor" matches are included) and recall (whether "good" matches are not included).

A search for "IBM" may be for the IBM Web site, but it could just as likely be for articles about IBM as a company, or articles with information on IBM-compatible PCs, etc. Typical searches are for information *about* the search term, and can be satisfied by any number of "*relevant*" items, any or all of which may be previously unknown to the searcher. In this sense there is no *specific* target object (page, document, record, etc.), only some open ended set of objects which may be useful with regard to the search term. The discovery search term does not signify a single intended object, but specifies a *term* (which is an attribute associated with one or more objects) presumed to lead to any number of relevant items. Expert searchers may use searches that specify the subject indirectly, to avoid spurious hits that happen to contain a more direct term. For example, searching for information about the book Gone With The Wind may be better done by searching for Margaret Mitchell, because the title will return too many irrelevant hits that are not about the book itself (but may be desired for some other task).

In other words, the general case of discovery searching that typical search engines are tuned to serve is one

where a search is desired to return some number, n, of objects, all of which are relevant. A key performance metric, *recall*, is the completeness of the set of results returned. The case of a signifier for an object, is the special case of n=1. Only one specific item is sought. Items that are not intended are not desired--their relevance is zero, no matter how good or interesting they may be in another context. The top DirectHit for "Clinton" was a Monica Lewinsky page. That is probably not because people searching for Clinton actually intended to get that page, but because of serendipity and temptation--which is a distraction, if what we want is to find the White House Web site.

In addition,

-CF obtains feedback from a group of users in order to serve each given user on an overall, non-contingent basis--without regard to the either the intent of the user at a specific time, or to being requested in a specific context.

-RF is used by a single user to provide feedback on their intent at a given time, but still with no presumed intent of a single target.

More broadly, searching techniques are generally not optimized based on using a descriptor which is also an *identifier*--they provide more generally for the descriptor to specify the nature of the content of the target, not its name. There are options in advanced search techniques which allow specification that the descriptor is actually an identifier, such as for

searching by title. Such options may be used to constrain the search when a specific target happens to be intended, but no special provision is made to apply feedback to exploit that particular relationship or its *singularity*.

Moreover, none of the currently available key word systems utilize heuristic techniques actually to determine the site intended by the user. Instead, the current systems teach away from such an approach by their use of registration, rather than user intention, to assign key words to map to web pages. Thus, the current techniques are not directed to solving the problem of finding the one, correct site for a particular signifier.

Thus, the need exists for a system that would enable a user to find a desired Web document by simply entering an intuitive key word or alias and that would perform a one to one mapping of the alias with the URL actually desired by the user, and which would use heuristic techniques to assist in providing the correct mapping, and improving system accuracy over time.

SUMMARY OF THE INVENTION

In consideration of the above deficiencies of the prior art, it is an object of the present invention to provide a method of signifier mapping that allows a user to locate to a particular network resource, in the preferred embodiment a web page, by simply entering a signifier or alias.

Thus, the present invention is generally directed to a technique for intelligent searching or matching where a *signifier* is given and is to be related to a name or address of an *intended* target object.

Signifier, in the context of the present invention means:

-an identifier, referent, or synonym for the name or address of a specific resource (a target object) presumed to exist in some domain; but

-not necessarily a "name" or "address"--a canonical identifier that has been assigned by some authority or pre-set by some convention (names are a subset of signifiers--those which are canonical or pre-established);

-not necessarily a description of content or subject matter (concepts or words);

-an identifier that has cognitive significance to the user, and presumed communication value in identifying the intended target object to another person or intelligent agent.

In addition, this cognitive/communication value is based on a perceived relationship (meant to have minimal ambiguity) to an identifier, which might be an assigned name or a name based on common usage, but which need not be exact, as long at it serves to signify the intended target.

More generally, descriptors may possibly be considered to be signifiers, if they are intended to be unique or minimally ambiguous (e.g. "the company that commercialized Mosaic" or "the company that sells the ThinkPad").

It is a further object of the present invention to provide a system in which heuristic techniques are used together with user feedback to improve the accuracy of signifier mapping.

None of the many solutions to the signifier mapping problem (Netword, Centraal, Goto, etc.) have identified learning as a valuable technique. This may be because what naturally come to mind are techniques based on pre-defined mappings that make the use of "de jure" explicit registration. That teaches away from the idea of trying to learn the mappings heuristically from colloquial usage. (The same applies to attempts at creating systems for "user friendly names" in other directory systems.) Since the mappings are understood as being defined or registered, why would one try to learn about them? But actually, the mappings are just like natural language—they are dynamic, evolving, and

ambiguous, and can only be resolved in terms of *learned usage within a context* - which is best addressed by learning, as in the present invention, not registration or other static mappings as appear in the prior art.

The use of heuristic, adaptive feedback-based techniques operates in significantly different ways when focused on signifier mapping, and this can be exploited by isolating such tasks. The key difference between the present invention and most common searching tasks is that in the prior searching techniques, there is no intention of a specific target object that is known to exist.

The present invention has several advantageous features, various combinations of which are possible:

1) a special purpose mapping engine for locating popular sites by guessed names;
2) automatic display of the target site (if located with reasonable confidence);
3) an optional simplified mode of direct entry of a guessed site name; and
4) use of user expectations, such as popularity of guesses intended for a given site, as a primary criterion for translating names to sites, with provision for protection of registered trademarks or other mandates.

In accordance with one aspect of the present invention, a finder or locator server is established. The server is configured to work with a user interface that allows

users to enter an guessed name or alias, as easily as if the user knew the correct URL for the intended target resource. In response to entry of the alias, the finder server accesses a database that includes, in a preferred embodiment, conventional Web-crawler-derived index information, domain name registration information, as well as user feedback from previous users of the server, and looks up the correct URL, i.e., the one URL that corresponds to the alias and causes the user's browser to go automatically to that URL, without the user having to view and click on a search results page, if the correct URL can be determined with a predetermined degree of confidence.

In one preferred embodiment, the server is structured to accept the alias as a search argument and do a lookup of the correct URL and the return of same to the browser, without the intermediate step of the user having to wait for and then click on a search results web page. The automatic transfer is preferably effected using standard HTML facilities, such as a redirect page or framing. Redirect is effected by placing pre-set redirection pages at the guess URL on the server. Alternately, the redirect page can be generated dynamically by program logic on the server that composes the page when requested.

The present invention advantageously uses feedback and heuristic techniques to improve the accuracy of the determination of the correct URL. If a suggested match is found by the look-up technique and the accuracy of

the mapping is confirmed by user feedback, then, after directing the user to the URL, the result is stored in the server to improve the accuracy of subsequent queries. The server database includes a list of expected terms and expected variants that can initially be catalogued to provide for exact matches. This list is updated by the learning processes discussed in more detail below.

If, on the other hand, a probable one intended match cannot be determined, the finder server preferably uses intelligent techniques to find a selection of links to possible matches ranked in order of likelihood, or could return a no-match page with advice, or a conventional search interface or further directories.

According to a preferred embodiment of the invention, each of the selection of links are configured not to go directly to the target URL. Rather, the links are directed back to a redirect server established by the finder server, with coding that specifies the true target, and feedback information. The finder server can in this way keep track of user selections.

In accordance with an advantageous aspect of the invention, such feedback information is used to improve the results of the search by promoting web sites almost universally selected to exact match status, and by improving the ranking of possible lists in accordance with which links are most often selected. Preferably, a confidence parameter can be generated from such

tracking to control whether to redirect to a URL or to present a possible list to users.

In furtherance of the above and other objects, there is provided, a designated server, accessible on the Internet, the designated server being configured to respond to relocation requests that specify an identifier, corresponding to a target resource, that may not be directly resolvable by standard Internet Protocol name resolution services to the URL of the target resource. In a direct entry embodiment of the present invention, requests are passed to the relocation server by sending a relocation URL that designates the relocation server as the destination node and appends the identifying information for the identifier as part of a URL string. The relocation server extracts the identifying information and translates it into a valid URL for the target resource. The relocation server is configured, in the event that a unique URL can be determined with respect to the target resource, to cause the target resource to be presented to the user without further action on the part of the user.

Preferably, the user requests are entered at a web browser, the relocation or search server determines the valid URL for the target resource by performing a look-up in a database, and the response from the relocation server is in the form of a redirect page that causes the user's web browser to obtain the target resource.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A is an architectural block diagram of a server computer system internetworked through the Internet in accordance with a preferred embodiment of the present invention;

Figure 1B is a flow diagram illustrating a method of obtaining feedback from multiple users to be applied in searching or signifier mapping;

Figure 2 is flow diagram showing a method of signifier mapping using feedback and heuristics to continually improve the performance of the mapping;

Figure 3 shows an example of a database entry for the finder server of the present invention;

Figure 4A is a flow diagram illustrating a technique of feedback weighting for probable results in signifier mapping;

Figure 4B is a flow diagram illustrating a technique of feedback weighting for possible results in signifier mapping; and

Figure 5 is a flow diagram illustrating how feedback is used in a preferred embodiment to discriminate a probable target resource in accordance with the present invention.

DETAILED DESCRIPTION OF THE DISCLOSED EMBODIMENTS

"Population cybernetics" and the Internet

As a general matter, the present invention relates to a technique that collects experience (a knowledge base) from a mass population that is open ended or universal, either over all domains, or over some definable subject or interest domain or strata. This represents a significant improvement over prior art techniques, which are generally limited in the scope of the population and extent of experience from which they draw their knowledge base.

The technique of the present invention, in a preferred embodiment, uses the Internet to do this in a way that is powerful, economical, and far-reaching. The technique, in the preferred embodiment, uses the Internet to enable collection and maintenance of a far more complete knowledge base than has been used with any prior technique except Collaborative Filtering (CF).

In the present invention feedback learning is advantageously utilized, so that the information is not just collected, but refined based on feedback on the accuracy of prior inferences.

In its broad sense the present invention constitutes a kind of "population cybernetics," in that the learning does not just collect a linear knowledge base, but uses

a feedback loop control process to amplify and converge it based on the results of prior inferences, and that it works over an entire population that is open, infinite, and inclusive. This is in contrast to prior learning techniques, which draw on necessarily finite, closed populations.

The use of population group information to achieve signifier mapping differs from the prior art technique of collaborative filtering in at least the following manner:

Whereas both CF and the technique of the present invention draw on knowledge of a population group to make inferences, CF obtains *ratings* of many things by many people to suggest other things (that may also be highly rated by the user, based on correlation with the group), and CF *does not involve* a *specific input request*, but rather seeks a *new, previously unknown item* in a category. On the other hand, the present invention obtains *translations* of many signifiers by many people to suggest the intended translation of a signifier and *involves a specific input request* to be translated to identify a *known intended target*

Although the technique of signifier mapping will occasionally be referred to loosely as searching, it is more accurately translation, because the target is intended and known, just not well specified. This differs from typical Web or document searching, which typically seeks unknown, new items.

The technique of the present invention also differs from natural language (NL) translation or understanding, in that the input is *atomic*, in that it [is] **has {***}** no context as part of a body of discourse (a text). NL understanding techniques on the other hand translate words as components of concepts embedded in texts having a *context* of related ideas. Thus the cues of context in a discourse are absent, and the translation must be done without any such cues (although knowledge of the user may provide a useful context of behavior, demographics, psychographics that has some value in inferring intent **{***} and knowledge of the user's prior requests may provide additional useful context information [this is more relevant to task/domain segmentation, but seems worth noting here, ~~unless there is reason to avoid introducing that~~ material]**). The task is to infer or predict intention, rather than to understand meaning, because there is no basis to infer meaning in any conceptual sense. The input is disjointed from any **{***} surrounding** context, and if not seen before (from the user or others), there is little useful information on either its meaning or intention. The present invention seeks to infer intention based on limited data, primarily the input request, and draws on group data (of request translations) as its strength.

The task of the present invention has similarities with cryptanalysis, in that both the present invention and cryptanalysis use data about communications behavior

from groups of communicators to make inferences.
However the task differs in that

- Cryptanalysis deals with intentional hiding of meaning or intention, where the technique of the present invention is applied to cases where the hiding (of the intention of a signifier) is not at all intended; and
- Cryptanalysis seeks to infer meaning (ideas) drawing on context in a discourse, like NL understanding, not usually to infer the intention of a signifier (of objects or actions) which is not in a context.

This point of intention versus meaning is subtle, but has to do with communication of commands or requests as opposed to concepts.

- One view of this is the idea of "requests," as opposed to declarations or assertions, in the use of language.
- This task of recognizing commands (vs. meanings) has parallels in the task of robot control, such as that based on spoken commands. The similarity is in training understanding of the speech of many users to **{***} be** speaker independent, and to infer meanings of a current speaker from that of others. The difference is that the tasks addressed in the present invention deal with a very wide, effectively infinite universe of commands (intended objects), while robot control techniques have generally been limited to very small sets of commands (partly because of the inability to apply mass experience).

Thus the technique of the present invention could be viewed as addressing a special class of robot control (in which experience data and feedback is accessible), and may ultimately be extensible to other robot control applications as such data becomes accessible over the network.

The social dimension is critical for inferences relating to shared objects or resources. Names draw on social conventions and shared usage. This social usage information is essential to effective mapping of signifiers to resources. De-jure naming systems can underlie a naming system, as for current Internet domain names, but de-facto usage is the essential observable source of information for fullest use. De-jure systems suffer from entropy, corruption and substitution, while de-facto usage is pragmatic and convergent to changing usage patterns.

This applies to a variety of name-able resources:
_ Web domain names;
_ Web sub-site names ({***} **such as** to find sub-areas);
_ People or business names;
_ Department, agent, or service identifiers (({***} **such as** to find
contact points);
_ Policy capability specifications (({***} **such as** to find permissions, such as someone who can provide access to a given resource for a given purpose, such

as confirming employment status or update-access to a report);

_ Information sets or collections (to find reference tools that are known to exist, such as an IBM dictionary of acronyms, or an index of papers in ACM publications);

_ Other robot control tasks, as social experience and feedback becomes accessible.

Social usage information can be combined with other sources of information in a heuristic fashion. For example, there could be a hierarchy that might be used in order, as available:

1. Personal defined usage information, such a defined personal nicknames;
2. Public de-jure defined mappings or directories;
3. Personal usage information (a person's own undefined nicknames, learned from ({***} [my] **that person's** own usage/feedback);
4. Social de-facto usage information;

This is just one possible sequence, but shows how the usage data can take searching beyond what has been defined.

As discussed above, a preferred embodiment of the present invention relates to a method and apparatus for locating a desired target resource located and accessible on a network, in response to user entry of a guessed name or alias. In illustrating the preferred embodiment, the apparatus is shown as a server

computer, or computers, located as a node on the Internet. However, the present invention is in no way limited to use on the Internet and will be useful on any network having addressable resources. Even more broadly, the present invention is useful for any similar task of identifying an intended target for an action in which automatic facilitation of that action is desired, where feedback from a large population can be obtained to learn whether a given response was in fact the one that was desired. Control of robots, as discussed herein, is one example of such broader application.

The finder server of the preferred embodiment of the present invention allows users to enter a guessed identifier or alias, as easily as if they knew the correct URL. Specifically, the finder server of the present invention accepts a guessed name, or alias, from a user, uses a look-up technique, enhanced by heuristics preferably taking into account previous users' actions, to determine a correct URL for the intended target resource, and causes the user's browser to go to that URL automatically. Preferably this is done without the added step of first viewing and clicking on a search-results page, where an initial search finds the intended target resource with a predetermined degree of certainty. Such a resource will be referred to hereinafter as a "probable". In accordance with a preferred embodiment of the present invention, this functionality can be implemented by:

- Publicizing the locator server under an appropriate URL name, for example, guessfinder.com.
- Setting up the server to, in response to entry of a guessed name or alias, do a lookup to the correct URL and return a response that causes the user's browser to go automatically to the specified URL. Such an automatic transfer can be effected using a standard HTML facilities, such as a redirect page, or framing.
- If the guess does not provide an exact match in the lookup phase, using feedback and heuristic techniques to create and present to the user a selection of links to possible matches. Alternately, the user may be presented with a nomatch page with advice, or directed to a conventional search interface, or further directories.

It is contemplated that the use of aliases for attempting to locate a web site associated with company name or brand name would be found useful. For example, the aliases "s&p", "s-p", "sandp", "snp", "standardandpoors", "standardnpoors", "standardpoors" should preferably all map to www.standardpoors.com. In addition to companies and brands, other important name domains would include publications, music groups, sports teams, and TV shows.

The present invention advantageously provides for learning and feedback on the basis of user preferences to automatically and dynamically build a directory of names and sites that maps to the actual expectations and intentions of a large population of users, and

adapts to changes over time, including the appearance of new sites, thus optimizing utility to them.

The finder server of the present invention effectively provides a secondary name space, administered by the organization operating the finder system, through the automated heuristic methods described here, that maps to, but is not dependent on, the URL name space. The finder site computer has access to a data base containing entries for any number of popular sites, with any number of likely guesses and variations for each site.

As a result of the service provided when the present invention is implemented, site sponsors could skip the cumbersome and costly process of obtaining specific mnemonic URLs or alternate URLs in many cases (especially with regard to domain names). Even with a number of conventional URLS, this service could be a supplement, for additional variations. The problem of pre-empted URL domain names would also be avoided, except where there is legitimate and significant pre-existing usage.

A key to utility is to be able to directly connect in response to most guesses, and ambiguities could be a limiting factor. To avoid that it is desirable to exploit Pareto's Law/the 80-20 rule and do a direct connect even when there is an uncertain but likely target. For that to be useful, it must be easy for users to deal with false positives.

Correction after arrival at a wrong site can be made relatively painless by allowing a subsequent request to indicate an error in a way that ties to the prior request and adds information. For example a request, guessfinder.com/lionking, that located the movie but was meant to find the play could be corrected by entering guessfinder.com/lionking/play. A more efficient coding might explicitly indicate an error, such as guessfinder.com/!/lionking/play. Even with the error, this would be quicker and easier than conventional methods.**{***} Note that this example was illustrated with the direct URL coding techniques described below. Similar post-arrival corrections can be made with other user interface techniques, such as a frame header that includes appropriate user interface controls to report feedback (much as conventional search engines allow for "refinement" of prior searches), also described below. [Note this also starts to approach task/domain segmentation, but seems basic enough to include here]**

Correction in-flight can be achieved by using the existing visibility of the redirect page, or enhancing it. When a redirect page is received by a user's browser, it appears for a short time (as specified with an HTML refresh parameter) while the target page is being obtained. In addition to **{***} affording a way to optionally present revenue-generating (interstitial)** advertising content, that page preferably lists the redirection target, as well as alternatives, allowing

the user to see the resolution in time to interrupt it. This is most useful with a browser that permits a redirect to be stopped in mid-stream by clicking the stop button, leaving the redirect page on display, and allowing a correct selection among alternative links to be made. **{***} Alternatively, a multi-frame (multi-pane) display could be used to allow a control frame to remain visible while the target page is loading in a results frame, as described below.**

Note some of the typical parameters and control points that would be relevant:

"New" sites.
Applies when the user wants a site but is provided neither a direct hit, nor a correct possible. Users would find the site via alternate means (offered through the service or not). The user then submits an add-site request, via the Web or e-mail. If the number of add-site requests over a set interval exceeds a set (low) threshold, the site is added as a possible, or a direct **{***}hit** if there are no competing alternatives. Such adds would be provisional, and could be dropped if requests are not sustained.

Possibles
Low confidence possibles would be listed low on the list, and selections would be tracked. If selections are strong, they move up the list. If selections are very weak, they would drop off after some interval. The threshold to add back sites that were dropped might

be higher for a time, to limit oscillation and false adds. If possibles are well ahead of alternatives by some threshold over some interval, they would be promoted to direct hits.

Direct hits

Feedback on false positives would be collected. This could be via links in frames, redirect pages, interstitials, or other means, as suggested previously. If false positives exceed a threshold, the site would revert to a possible and the common alternatives would be listed as well.

Parameter issues: thresholds, intervals, smoothing, damping, overrides.

Basic parameters include the various thresholds and time intervals for measurement. Smoothing techniques (such as exponential smoothing) would be applied to adjust for random variations and spikes, to improve forecasting. Damping mechanisms could be used to limit undue oscillation from state to state. Overrides would provide for mandated or priority matches, such as for registered trademarks, on either a weighted or absolute basis, as appropriate.

Figure 1A illustrates a first embodiment of the present invention, as implemented on the Internet. The finder server 10 includes a computer or computers that perform processing, communication, and data storage to implement the finder service. Finder server 10

includes a finder processing/learning module 101. Module 101 performs various processing functions, and includes a communication interface to transmit and receive to and from the Internet 12, as well as with database 102, and is programmed to be operable to learn from experiential feedback data by executing heuristic algorithms. Database 102 stores, in a preferred embodiment, indexes of URL data that would allow the module 101 to locate, with a high degree of confidence, a URL on the Web that is an exact match for a target resource in response to a user's entry of an alias or guessed name. Preferably, the indexes store, in addition to available URL information, such as domain name directories, information relating to the experience of the server in previous executions of the finder service. As the server gains experience and user feedback, heuristic techniques are applied by module 101 to enable the returned URLs to conform more and more accurately to user expectations.

Users 11_0-11_N can access the Internet 12 by means of client computers (not shown) either directly or through an Internet service provider (ISP).[.]{***} As has been discussed previously, to make use of the present invention the user enters a guessed name, or alias, into his computer's browser and submits a query containing the alias to the finder server. The World Wide Web 14 includes computers supporting HTTP protocol connected to the Internet, each computer having associated therewith one or more URLs, each of which forming the address of a target resource. Other

Internet information sources, including FTP, Gopher and other static information sources are not shown in the figure.

The finder server includes operating system servers for external communications with the Internet and with resources accessible over the Internet. Although the present invention is particularly useful in mapping to Internet resources, as was discussed above, the method and apparatus of the present invention can be utilized with any network having distributed resources.

Entry of the alias by a user may be accomplished in a number of ways. In one embodiment, a usage convention can be publicized for passing the alias to the server within a URL string, such as guessfinder.com/get?ibm, for example, for trying to find the web page corresponding to the alias "ibm". In this case, the server is programmed to treat the string "ibm" as a search argument and perform the appropriate processing to map the alias to the intended target resource. A similar effect can be obtained by the somewhat simpler form guessfinder.com/ibm, if the server is programmed appropriately. Alternately, the user can visit the web site of the finder server and be presented with a search form, as is done in conventional search engines. A third option is to provide a browser plug in that allows direct entry of the key word in the browser's URL window or any alternative local user interface control that will then pass the entry on as a suitably formatted HTTP request.

It also would be preferable for an enhanced user interface to be phased in as the service gains popularity. This preferably would be accomplished by a browser plug-in, or modifications to the browser itself, to allow the alias to be typed into the URL entry box without need for the service domain name prefix (such as, guessfinder.com/...). Instead, such an entry would be recognized as a alias, not a URL, and the prefix would be appended automatically, just as http://... is appended if not entered with a URL in current browsers.

Figure 1B is a flow diagram illustrating a technique for obtaining and learning from feedback responses gathered from a large group of people, in the example, users 1, 2, . . . n. Such a technique can be used in a variety of applications, and in particular in traditional search engines, or in mapping to identify particular web sites, as in alias or signifier mapping.

In Figure 1B, users 1, 2, . . . n represent a large community of users. In the flow diagram, the flow of query items from the users is indicated by a Q, the flow of responses back to the users is indicated by an R, and the flow of feedback results provided by the users' actions, or responses to inquiries, is indicated by an F. As can be seen from the figure, Query (a, 1) is transmitted from user 1 to the service 2, which can either be a searching or a mapping service. The service has learning processor 4, which interfaces with

a database 6. The database 6 contains, among other things, indexes and feedback information gathered from previous queries. In response to the query, the user 1 is provided with a response R(a, 1). User 1 then is provided with the opportunity to transmit user Feedback (a, 1) to the Service 2. Learning processor 4 stores the feedback information in the database 6, and is programmed with one or more heuristic algorithms enabling it to learn from the feedback information to improve the returned search or mapping results. The feedback provided will improve the results offered, for example by positively weighting results preferred by users, so that, over time, more accurate results can be obtained.

Figure 2 is a diagram illustrating the logical flow used in applying the general technique of learning from user feedback shown in Figure 1 to signifier mapping, in accordance with a preferred embodiment of the present invention. A user enters a Query consisting of a signifier, represented by Q^s. The server, in response to receipt of the query, parses the query, at step S02, and in step S04 performs a database lookup in an attempt to determine, if possible, the exact target resource intended by the user. Database 6 includes index data as well as feedback data obtained from users in previous iterations of the signifier mapping program, is accessed. The stored data structure is described in more detail below.

In step S06, the program discriminates a probable intended target making use of the index information such as domain registration indexes, and other resources, as well as the feedback information stored in the database. In step S08, if a likely hit, or exact match has been identified, that is, a web page has been located with a high confidence parameter, the flow continues to step S10. At step S10, a direction is prepared to the likely hit URL. A list of alternatives optionally may be provided for presentation to the user at the same time, in case the likely hit turns out not to be the target identifier. At step S12, the server sends information R^s to the user, more particularly to the user's browser, to effect a link to the likely hit. Optionally, the alternate list is also provided at the same {***}time[?].

In step S14, the viewed page is monitored by the server and the user, by his actions, provides feedback. Most readily determined with no assistance from the user is the fact of the user having chosen the link. This may be determined, for example by a redirect, in which an intermediate server is transparently interposed between the browser and the target page, and thus able to identify the user and the URL target based on coding built into the URL that the user clicks. Also desirable is the amount of time the user spends at the site, which will be an indicator of whether the site is the intended target. This may be ascertained, for example, if clickstream data can be obtained, such as

through the use of a monitor program that works as a browser add-in or Web accessory, such as the techniques offered by Alexa. Other feedback can be provided by asking the user. This can, for example, be done conveniently by using a small header frame served by the relocation service that appears above the actual target page, and that includes controls for the user to indicate[d]{***} whether or not the results were correct. The URL of the viewed page is recorded, together with any other feedback, for use in improving the accuracy of subsequent iterations of signifier mapping. At step S26, the feedback data is supplied to a feedback weighting algorithm, described in detail below, which generates appropriate weighting factors to be stored in the database for use in subsequent mappings.

If it is determined at step S08 that the result is not a likely hit, the flow proceeds to step S18, where a list of the top m hits (m being a predetermined cutoff number), preferably drawing on the list of possible hits from a conventional search engine, or by employing the same techniques as a conventional search engine, is prepared. Unlike conventional search engines, the ranking of these hits is based primarily on experience feedback data as described below. In addition, where such feedback is limited or absent, it would be supplemented by variants of more conventional search engine weighting rules that **{***}are** expressly tuned to the task of finding a single intended result (i.e., high relevance by low recall) rather than many results

(high relevance plus high recall). The list is presented, at step S20, to the user as R^s. The user, by the selections made from the provided list, and from other feedback, such as how long the user spends at each link, supplies feedback to the system. This information F^s is monitored, at step S22 and recorded, at step S24. The recorded information is supplied to the feedback weighting algorithm, at step S26, the output of which is stored in the database for use in subsequent iterations of the signifier mapping.

Figure 2 illustrates the simple case in which a user is directed to a target URL if the target has been determined to be a probable hit, and is presented with a list to choose from if the target cannot be identified with sufficient certainty. However, it is well within the intended scope of the invention for alternate methods to be employed. For example, the user interface (UI) could be extended, either by framing, or a browser plug-in or extension, to provide multi-pane/multi-window results that allow a pane for each type of response, e.g., the target response and a list of possibles, regardless of the level of confidence in the result. In such a case, the format for presentation of results would be the same whether a probable has been located or not, but the learning from feedback and ranking would still seek to determines "correctness" based on the varying feedback cases.

Figure 3 illustrates a preferred method of organizing index data to allow for storing and updating of the

most probable hits for a given query. As can be seen from the illustration, for each query, whether single element queries or compound queries, there is stored a list of associated possible targets. Linked to each of these query/target pairs is a raw score, an experience level, and a probability factor. As feedback enters the system, the index data is updated to reflect the user feedback. The updating process will be described below. While the index shows preferred weighting criteria, these are only a sample of the kind of criteria that can be correlated to the query/target pairs. In a simple embodiment, the raw score would be based only on selections of hits, and explicit feedback **{***} on** [and] correctness as described below. Other embodiments could add feedback data on time spent at a target. Additional variations would include weighting based on the recency of the feedback, and on the inclusion of non-feedback data, such as the various syntactic and semantic criteria used for relevance weighting by conventional search engines.

The process of maintaining the guess-target database is adaptable to a high degree of automation, and this can be highly responsive to new sites. An outline of such a method is:

All guesses are logged and analyzed.

Ambiguous hits are tracked as described earlier.

Complete non-matches are sorted by frequency to identify common new requests (in real time). Changes in ambiguous match patterns could also flag appearance of new sites.

Common new requests preferably are fed to an automated search tool that would use existing search engines, hot site lists, and name registration servers, etc. to identify possible targets.

Automated intelligent analysis of those results can seek to qualify probable targets.

High confidence (or possible) targets preferably are added, and then tracked based on the feedback mechanism described earlier, in order to self-correct. A confidence parameter preferably is used to control whether to redirect or to present a possibles list to users.

Human review and correction also preferably is used to supplement this.

Figure 4A illustrates a preferred technique for weighting the results using feedback data for hits that have been determined to be probable hits. In step S30, if the user feedback from the probable result indicates that the probable was in fact the target URL the user was searching for, the flow proceeds to step S32 where the raw score for that query/target pair is incremented by $factor_Y$. If the user returns feedback indicating that the probable was not the target resource the user had in mind, the flow proceeds to step S34 where the raw score for that query/target pair is decremented by $factor_N$. If the user provides no feedback, then the flow proceeds to step S36 where the raw score is decremented by $factor_O$, which can be zero. After execution of any of steps S32, S34 or S36, the flow

proceeds to step S38, at which the experience level score is incremented by $Efactor_C$.

Figure 4B illustrates a preferred technique for weighting in accordance with user feedback in the case of possibles, i.e., items on the list presented to the user when no probable result can be located. As shown in the figure, if a possible is selected by the user from the presented list, at step S40, the fact of selection is recognized, preferably by use of a redirect server that allows the system to keep track of which link was chosen. Additionally, the amount of time the user spends at the selected link may be ascertained. Making use of the information gathered in the redirect and such other feedback as may be obtained, the raw score for the query/target pair is incremented, at step S44, by $factor_S$. The user is then requested to provide additional feedback after the user has finished viewing the link.

In a preferred embodiment of the present invention, this feedback is gathered from the user by presenting the user with a frame that includes a mechanism, such as a check box, or radio button, that allows the user to indicate whether the selected possible was in fact the intended or "correct" target resource. If it is determined, at step S42, from the feedback that the link was the correct target, the flow proceeds to step S46, where the raw score for that query/target pair is incremented by $factor_Y$. If the user returns a negative response, the raw score of the pair is decremented at

step S48 by a by $factor_{N'}$. If no feedback is received, the raw score is decremented, at step S50, by $factor_{O'}$, which can be zero. After execution of any of steps S44, S46, S48 or S50, the flow proceeds to step S32, at which the experience level score is incremented by $Efactor_{PS}$ in the case of selection of the link, and by $Efactor_{PC}$ if the link was the correct.

Figure 5 illustrates a detail of how the present invention ranks and discriminates a probable target. At step S100 a list of possibles is obtained. Next, the list is ranked, at step S102, on the basis of the expected probability as the target. In step S102, a discrimination criteria is calculated and compared with a predetermined threshold parameter. For example, if ProbTi is the expected probability that Ti is the correct target, a formula such as the example shown can be used to determine whether T1 stands out as more probable than T2 by a relative margin that exceeds a set threshold needed to judge it as the probable intended one target. When the threshold is not exceeded, the implication is that one of the secondary possibilities may very well be the intended one, and that directing the user to the slightly favored target may not be desirable.

In the preferred embodiment, when a link on a list of possibles is selected by the user, rather than connect the user immediately to the chosen link, the finder server first redirects the user to a redirect server where feedback data relating to the selection can be

gathered. One item of feedback that may be obtained in this manner is the very fact of the selection. Further feedback can be obtained by additional means, such as monitoring how long the user spends at the selected link, and by directly querying the user.

The redirect linking technique uses the target URL as a server parameter within a composite URL to control the intermediate server parameter within the URL to control the intermediate server. The target URL is embedded as a server parameter within a URL that addresses the redirect server, and the URL parameter is used to control the intermediate server process. Thus a server is called with a first URL, a redirect URL, that specifies the second URL, i.e., the target URL, as a parameter. For example

http://redirector.com/redirector?query12345678/targetserver.com/targetpath1/targetpage1.htm

where redirector.com is the intermediate server URL, query12345678 is a unique identifier of the user-query combination, and targetserver.com/targetpath1/targetpage1.htm is the target URL. The network ignores the parameter portion of the URL, which is passed as data to the server. The server acts on the parameter to perform desired intermediary processing, in this case, the logging of the fact that this link was clicked in response to query12345678, and to redirect the user to the intended location specified by the second URL. The token query12345678 could be a unique identifier

corresponding to a logged user-query entry, or it could be the actual query string.

The delay required for the redirect provides the opportunity for the display of interstitial advertisements. In addition, additional user feedback can be solicited during the delay, and the connection to the targeted URL can be aborted if the user indicates that the target site is not the one he or she intended. In addition to using the redirect when a link is selected, the technique also preferably is used when an exact match is found, to provide a brief delay before connecting the user to the exact match, to present advertisements to give the user the time to abort the connection. In any event, the user preferably is given the opportunity to provide feedback after connecting to any site, whether directly as a result of an exact match, or as a result of selecting from a linked possibles list.

The redirect server of the present invention allows data to be gathered on each link as it is followed and redirected. The redirect link can be created in a simple static HTML. However, it is preferable to create the link dynamically for each user selection.

The finder is setup to recognize the feedback function, possibly as a CGI or other gateway/API function, and invoke the appropriate function to parse the URL or other data (referer, cookies, etc.), extract the target URL and feedback information for processing, and return

a page containing a redirect (or use framing or other means) to take the user to the desired target.

This mechanism is general, and can be used for many purposes. In the case of the finder server:

-Reasonably complex feedback information can be obtained, which at minimum would include the original guess. Thus a log of each guess that was not clearly resolved, paired with the corresponding user-selected target, can be obtained.

-That set of selected guess/target pairs can then be used to adjust the confidence levels in the guess/target database. Similar data on directly resolved pairs would also be applied, along with any data from wrong-match reports.

Other applications are to any situation where links go to sites other than the source. This would include results of conventional search engines, as well as resource directories, sites referring users to suppliers, advertisers, etc.

The above embodiments of the present invention have been described for purposes of illustrating how the invention may be made and used. The examples are relatively simple illustrations of the general nature of the many possible algorithms for applying feedback data that are possible. However, it should be understood that the present invention is not limited to the illustrated embodiments and that other variations

and modifications of the invention and its various aspects will become apparent, after having read this disclosure, to those skilled in the art, all such variations and modifications being contemplated as falling within the scope of the invention, which is defined by the appended claims.

WHAT IS CLAIMED IS:

{suggested wordings--maybe we need to discuss}

1. A method of finding, in response to entry by a user of a [guessed] resource [name] **{***}identity signifier,** a single **{***}intended[?]**[desired] target resource intended **{/meant?}** by the user to uniquely correspond to the [guessed] resource [name] **{***}identity signifier,** among a plurality of resources located on a network comprising a plurality of interconnected computers, the method for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method **{***} by a multiplicity of users {this may better go in a dependent claim for tactical reasons, but we may need it here to distinguish clearly from prior art}**
; and (b) a learning system structured to access and learn from information contained **{***} in** [on] the database, the method comprising:

receiving a [guessed] resource [name] **{***}identity signifier** from the user;

accessing the database to determine which, if any, of the indexed resources is **{***} likely to be the intended** [desired] target resource;

{* should this be dependent--seems secondary to mapping the signifier, which seems novel in itself:}** directing a computer of the user so as to [cause]**enable {*** want to allow both directly cause or enable by user click, etc.}** that computer to connect the user to

the address, if any, of the resource determined to be the target resource.

{*corresponding changes to all other claims? Or at least 1-5?}**

2. A method according to Claim 1, wherein a resource is determined at the accessing step to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

3. A method according to Claim 2, wherein if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

{* add a claim 3a, covering the case where both 2 and 3 are done, such as with multiple frames (possibly regardless of confidence). Similarly for 7+8 and 12+13, etc.}**

4. A method according to Claim 3, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

5. A method according to Claim 1, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed

identifier and the address of the resource to which his or her computer was directed.

6. An apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database;

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

7. An apparatus according to Claim 6, wherein a resource is determined to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

8. An apparatus according to Claim 7, wherein the apparatus is operable to, if the guessed name does not map uniquely to any of the indexed resources, to a predetermined confidence level, perform the following steps:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

request that the user select a link; and

cause the user's computer to go to the selected link, if any.

9. An apparatus according to Claim 8, wherein the apparatus is operable to, if a link has been selected, perform the following steps:

add information regarding the selection of the link to the feedback information in the database;

solicit user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, increment the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

10. An apparatus according to Claim 6, the apparatus being further operable to:

solicit user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, increment the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrement the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

11. A system for finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the system comprising:

finder server means having: (a) database means for storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system; and (b) learning system means for accessing and learning from information contained on the database;

receiving means for receiving a guessed resource name from the user;

accessing means for accessing the database means to determine which, if any, of the indexed resources is the desired target resource;

directing means for directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined in the access means to be the target resource.

12. A system according to Claim 11, wherein a resource is determined by the access means to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

13. A system according to Claim 12, further comprising:

presenting means for, if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting means for requesting that the user select a link from the presented list; and

means for causing the user's computer to go to the selected link, if any, upon a user selection of a link.

14. A system according to Claim 13, further comprising:

adding means for, if a link has been selected, adding information regarding the selection of the link to the feedback information in the database;

soliciting means for soliciting user feedback with regard to the selected link; and

means for, if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

15. A system according to Claim 11, further comprising:

soliciting means for soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

means for, if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence

factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

16. A computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method comprising:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

17. A computer-readable medium according to Claim 16, wherein a resource is determined in the accessing step to be the target resource if the database information indicates that there is at least a predetermined confidence level that the guessed name maps uniquely to the address of the resource.

18. A computer-readable medium according to Claim 17, wherein if the guessed name does not map uniquely to any of the indexed resources, to at least the predetermined confidence level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence factor, the resources having the highest confidence factors being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

19. A computer-readable medium according to Claim 18, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the selected link.

20. A computer-readable medium according to Claim 16, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the guessed identifier, incrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the guessed identifier, decrementing the confidence factor associated with the mapping between the guessed identifier and the address of the resource to which his or her computer was directed.

21. A system for finding resources on a network of interconnected computers on which a plurality of resources reside, the system comprising:

a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and

a finder server having:

(a) a database including: (i) an index of resources available on the network; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and

(b) a learning system operable to access and learn from information contained on the database,

the finder server being operable to locate, in response to entry by the user of a guessed resource name, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by:

receiving a guessed resource name from the user;

accessing the database to determine which, if any, of the indexed resources is the desired target resource;

directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

{* add similar parallel claim(s) covering the broader application: "Even more broadly, the present invention is useful for any similar task of identifying an intended target for an action in which automatic facilitation of that action is desired, where feedback from a large population can be obtained to learn whether a given response was in fact the one that was desired."?}**

ABSTRACT OF THE DISCLOSURE

An apparatus for finding resources on a network comprises a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding feedback gathered from a user of the apparatus in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database. The finder server is operable to locate, in response to entry by the **{***} any of a multiplicity of users** of a guessed resource name **{***}or identity signifier**, a single desired target resource intended by the user to uniquely correspond to the guessed resource name, from among a plurality of resources located on the network, by: receiving a guessed resource name from the user; accessing the database to determine which, if any, of the indexed resources is the desired target resource; directing a computer of the user so as to cause that computer to connect the user to the address, if any, of the resource determined to be the target resource.

FITZPATRICK, CELLA, HARPER & SCINTO

30 ROCKEFELLER PLAZA
NEW YORK, NY 10112-3801
212-218-2100
FACSIMILE (212) 218-2200

WASHINGTON OFFICE
1900 K STREET, N.W.
WASHINGTON, D.C. 20006-1110
(202) 530-1010
FACSIMILE (202) 530-1055

CALIFORNIA OFFICE
650 TOWN CENTER DRIVE, SUITE 1800
COSTA MESA, CALIFORNIA 92626-1925
(714) 540-8700
FACSIMILE (714) 540-9823

WRITER'S DIRECT DIAL NUMBER
(212) 218-2223

March 16, 2000

VIA FEDERAL EXPRESS

Mr. Richard Reisman
20 East 9th Street
Apt. 14K
New York, NY 10003

Re: New Application for Community Usage Learning

Dear Dick:

I enclose a final draft specification, claims and drawings for your invention Method And Apparatus For Utilizing User Feedback To Improve Signifier Mapping.

REDACTED

REDACTED

Sincerely yours,

Joe

Joseph W. Ragusa

cc: Bruce G. Bernstein, Esq.

NY_MAIN 69220 v 1

FIG. 1A

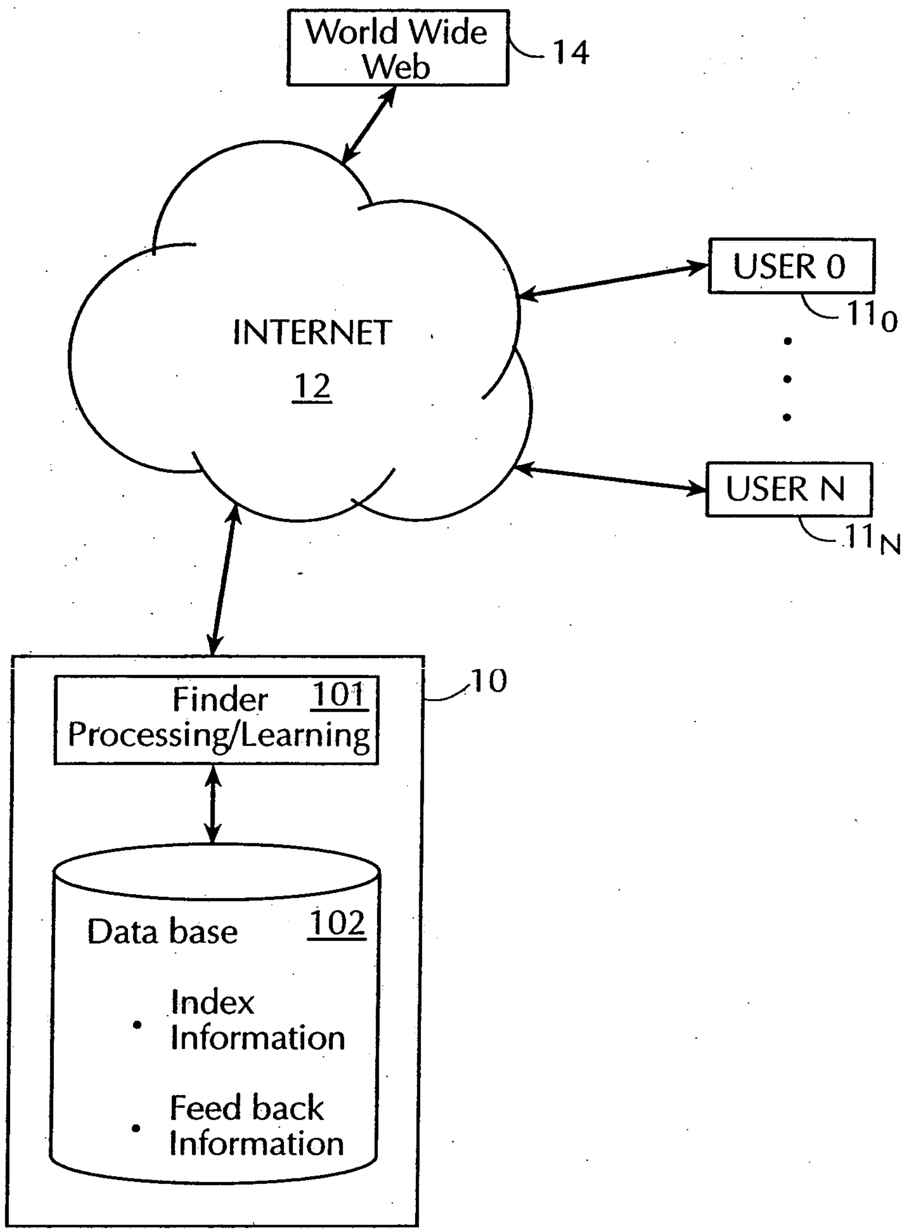
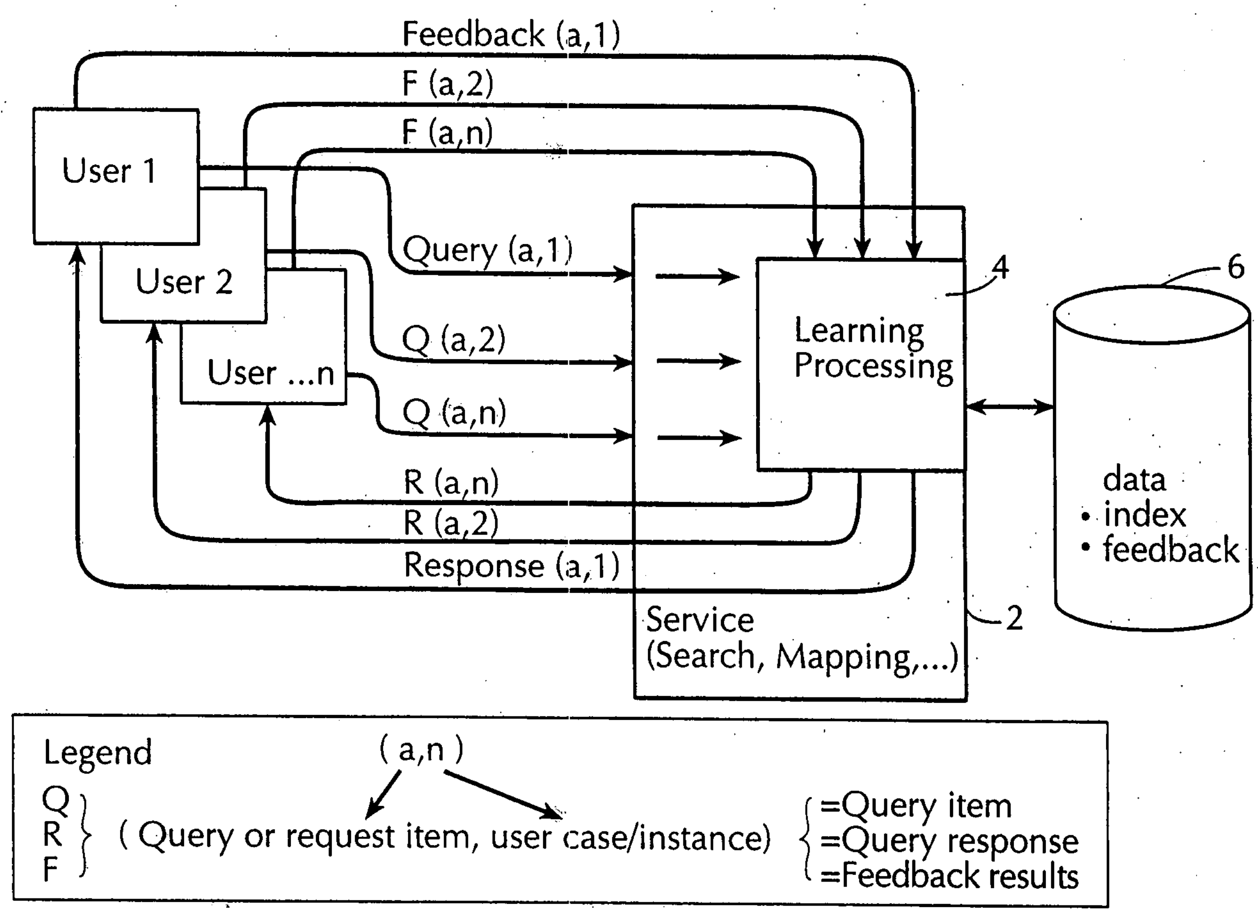
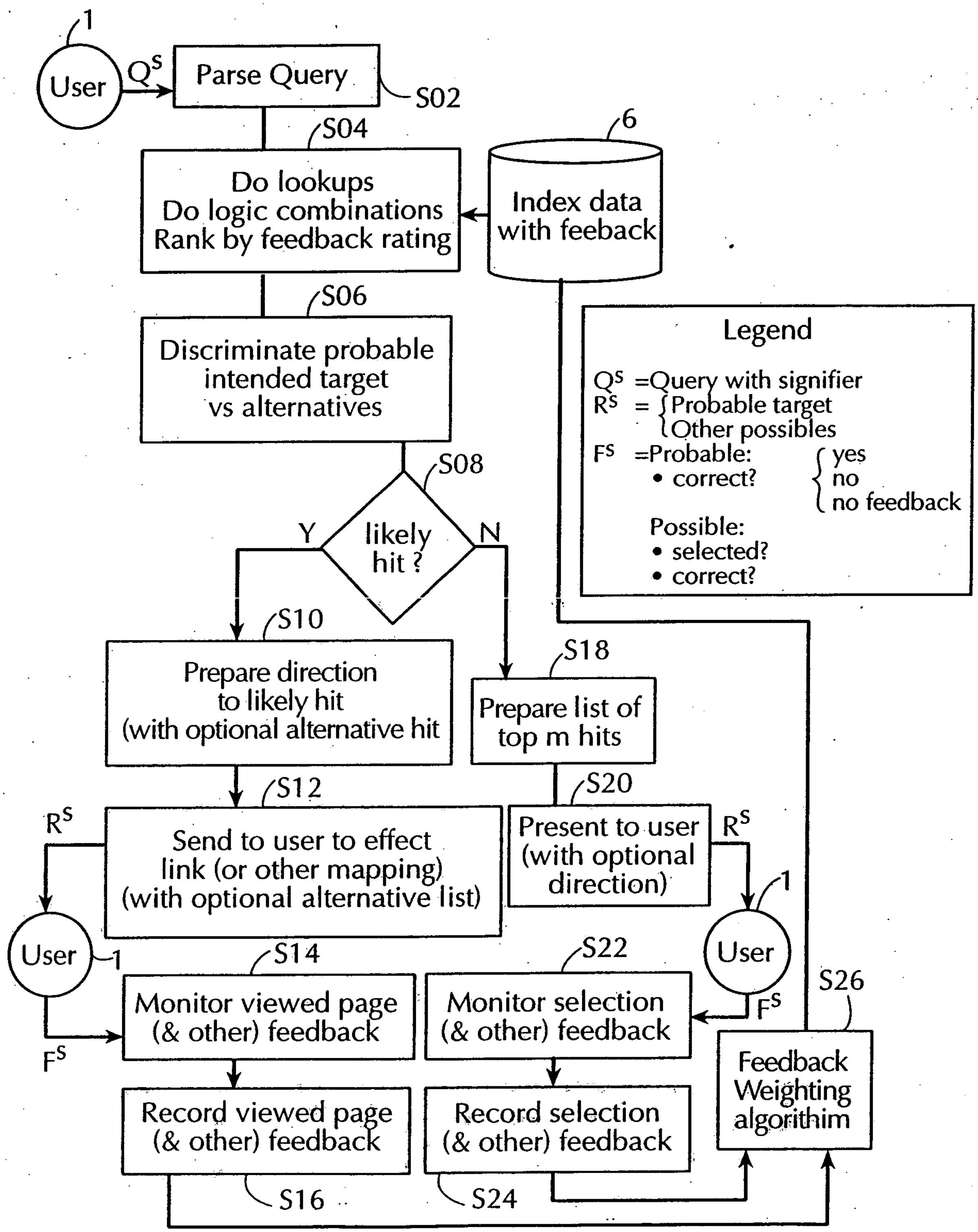

World Wide Web
14
INTERNET
12
USER 0
11_0
USER N
11_N
10
Finder 101
Processing/Learning
Data base 102
• Index Information
• Feed back Information

FIG. 1B



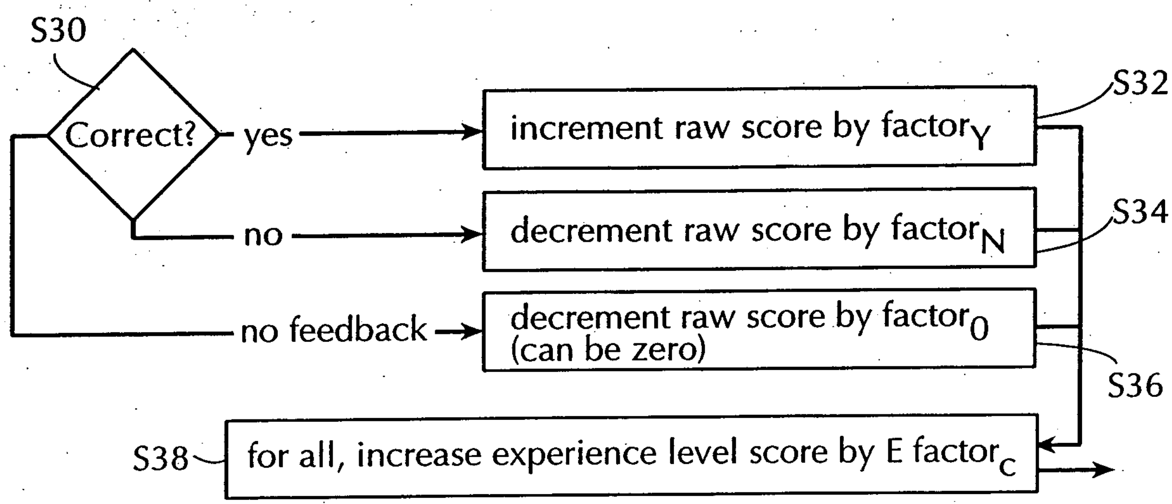
Feedback (a,1)
F (a,2)
F (a,n)
User 1
User 2
User ...n
Query (a,1)
Q (a,2)
Q (a,n)
R (a,n)
R (a,2)
Response (a,1)
Learning Processing
4
6
data
• index
• feedback
Service (Search, Mapping,...)
2
Legend
(a,n)
Q
R
F
(Query or request item, user case/instance)
=Query item
=Query response
=Feedback results

FIG. 2



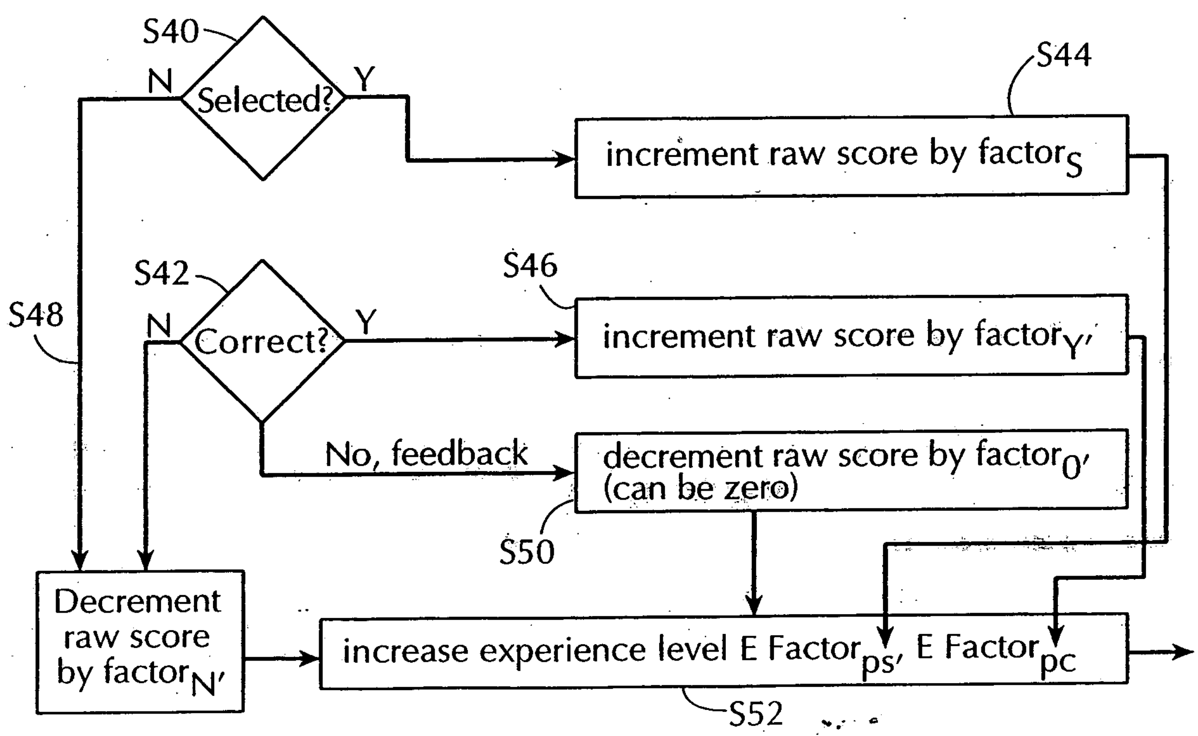
1
User
Q^S
Parse Query
S02
S04
6
Do lookups
Do logic combinations
Rank by feedback rating
Index data
with feeback
S06
Discriminate probable
intended target
vs alternatives
Legend
Q^S =Query with signifier
R^S = Probable target
Other possibles
F^S =Probable:
• correct?
yes
no
no feedback
Possible:
• selected?
• correct?
S08
Y
likely
hit ?
N
S10
Prepare direction
to likely hit
(with optional alternative hit
S18
Prepare list of
top m hits
S12
S20
R^S
Send to user to effect
link (or other mapping)
(with optional alternative list)
Present to user
(with optional
direction)
R^S
1
User
1
S14
S22
User
S26
Monitor viewed page
(& other) feedback
Monitor selection
(& other) feedback
F^S
F^S
Feedback
Weighting
algorithim
Record viewed page
(& other) feedback
Record selection
(& other) feedback
S16
S24

FIG. 4A



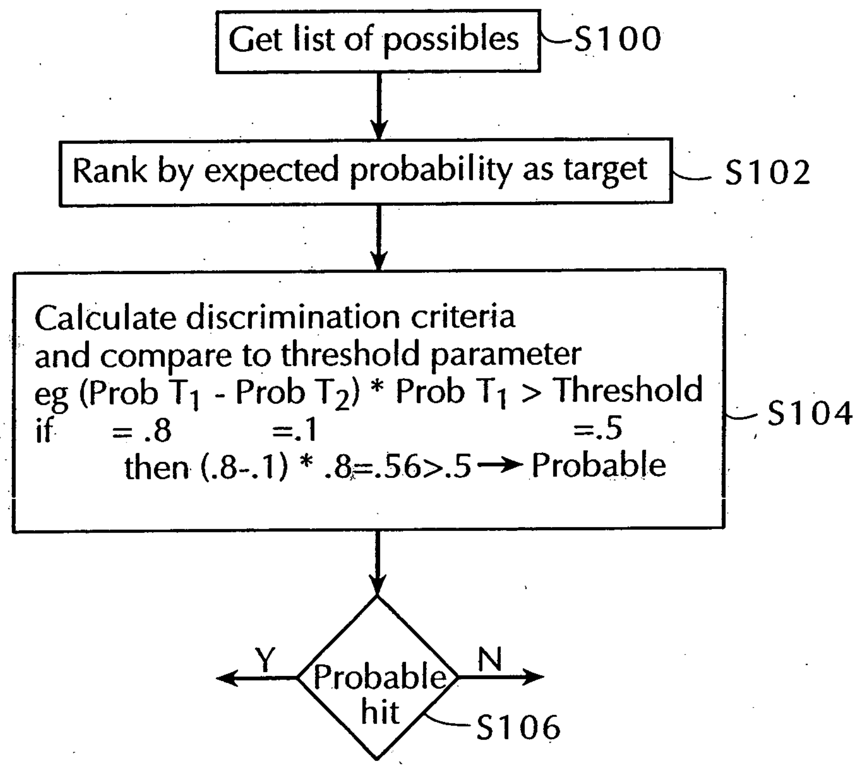
S30
Correct?
yes
no
no feedback
S32
increment raw score by factor$_{Y}$
S34
decrement raw score by factor$_{N}$
decrement raw score by factor$_{0}$
(can be zero)
S36
S38
for all, increase experience level score by E factor$_{c}$

FIG. 4B



S40
N
Selected?
Y
S44
increment raw score by factor$_{S}$
S42
N
Correct?
Y
S46
increment raw score by factor$_{Y'}$
S48
No, feedback
decrement raw score by factor$_{0'}$
(can be zero)
S50
Decrement
raw score
by factor$_{N'}$
increase experience level E Factor$_{ps'}$, E Factor$_{pc}$
S52

FIG. 5


Get list of possibles
S100
Rank by expected probability as target
S102
Calculate discrimination criteria
and compare to threshold parameter
eg (Prob T1 - Prob T2) * Prob T1 > Threshold
if = .8 =.1 =.5
then (.8-.1) * .8=.56>.5 → Probable
S104
Y
Probable hit
N
S106

TITLE

METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is directed to a computer-implemented product for locating and connecting to a particular desired object or target resource from among plural resources resident at distributed locations on a network.

Description of the Related Art

The worldwide network of computers known as the Internet evolved from military and educational networks developed in the late 1960's. Public interest in the Internet has increased of late due to the development of the World Wide Web (hereinafter, the Web), a subset of the Internet that includes all connected servers offering access to hypertext transfer protocol (HTTP) space. To navigate the Web, browsers have been

developed that give a user the ability to download files from Web pages, data files on server electronic systems, written in HyperText Mark-Up Language (HTML). Web pages may be located on the Web by means of their electronic addresses, known as Uniform Resource Locators (URLs).

A URL uniquely identifies the location of a resource (web page) within the Web. Each URL consists of a string of characters defining the type of protocol needed to access the resource (e.g., HTTP), a network domain identifier, identification of the particular computer on which the resource is located, and directory path information within the computer's file structure. The domain name is assigned by Network Solutions Registration Services after completion of a registration process.

While the amount of information available on the Web is enormous, and therefore potentially of great value, the sheer size of the Web makes the search for information, and particular web sites or pages, a daunting task. Search engines have been developed to assist persons using the Web in searching for web pages that may contain useful information.

Search engines fall into two major categories. In search engines falling into the first category, a service provider compiles a directory of Web sites that the provider's editors believe would be of interest to users of the service. The Yahoo site is the best known example of such a provider. Products in this category

are not, strictly speaking, search engines, but directories, and will be referred to hereinafter as “editor-controlled directories”. In an editor-controlled directory, the developer of the directory (the “editor”) determines, based upon what it believes users want, what search terms map to what web pages.

The other major category, exemplified by Altavista, Lycos, and Hotbot, uses search programs, called "web crawlers", "web spiders", or "robots", to actively search the Web for pages to be indexed, which are then retrieved and scanned to build indexes. Most commonly, this is done by processing the full text of the page and extracting words, phrases, and related descriptors (word adjacencies, frequencies, etc.). This is often supplemented by examining descriptive information about the Web document contained in a tag or tags in the header of a page. Such tags are known as “metatags” and the descriptive information contained therein as “metadata”. These products will be referred to hereinafter as “author-controlled search engines,” since the authors of the Web documents themselves control, to some extent, whether or not a search will find their document, based upon the metadata that the author includes in the document.

Each type of product has its disadvantages. Author-controlled search engines tend to produce search results of enormous size. However, they have not been reliable in reducing the large body of information to a manageable set of relevant results. Further, web site authors often attempt to skew their site's position in

the search results of author-controlled search engines by loading their web site metatags with multiple occurrences of certain words commonly used in searches.

Editor-controlled directories are more selective in this regard. However, because conventional editor-controlled directories do not actively search the web for matches to particular search terms, they may miss highly relevant web sites that were not deemed by the editors to be worthy of inclusion in the directory. Also, it is possible for the editor to "play favorites" among the multitude of Web documents by mapping certain Web documents to more search terms than others.

Recently, search engines such as DirectHit (www.directhit.com) have introduced feedback and learning techniques to increase the relevancy of search results. DirectHit purports to use feedback to iteratively modify search result rankings based on which search result links are actually accessed by users. Another factor purportedly used in the DirectHit service in weighting the results is the amount of time the user spends at the linked site. The theory behind such techniques is that, in general, the more people that link on a search result, and the longer the amount of time they spend there, the greater the likelihood that users have found this particular site relevant to the entered search terms. Accordingly, such popular sites are weighted and appear higher in subsequent result lists for the same search terms. The Lycos search engine (www.lycos.com) also uses feedback, but only at the time of crawling, not in

ranking of results. In the Lycos search engine, as described in U.S. Patent No. 5,748,954, priority of crawling is set based upon how many times a listed web site is linked to from other web sites. This idea of using information on links to a page was later exploited by the Clever system developed in research by IBM, and the Google system (www.google.com), which do use such information to rank possible hits for a search query.

Even leaving aside the drawbacks discussed above, search engines of both categories are most useful when a user desires a list of relevant web sites for particular search terms. Often, users wish to locate a particular web site but do not know the exact URL of the desired web site. Conventional search engines are not the most efficient tools for doing this.

Moreover, naming and locating particular sites on the Web is currently subject to serious problems. For example, appropriate names, including existing company names or trademarks, may not be available, because someone registered them first. Names may be awkward and not obvious, because of length, form/coding difficulties or variant forms, and names may not justify a separate domain name registration for reasons of cost and convenience, such as movie titles or individual products.

This problem results from a mismatch between the present network addressing scheme based on Uniform Resource Locators (URLs), which meet the technical

needs of the Internet software, and the needs of human users and site sponsors for simple, user-friendly mnemonic and branded names. This problem is largely hidden in cases where a user finds a site by clicking a pre-coded link (such as after using a search engine), or by using a saved bookmark. However, the problem does seriously affect users wishing to find a site directly, or to tell another person how to find it. To do this, the person must know and type the URL into his Internet browser, typically of the form sitename.com or www.sitename.com. Site sponsors are also seriously hampered by this difficulty in publicizing their sites.

Further, the current method of naming and locating Web sites has serious, widely known problems. Web site locator "domain" names are often not simple or easily remembered or guessed, and often do not correspond to company, trademark, brand or other well-known names.

As a result of the foregoing, site URLs (or domain names) are not intuitively obvious in most cases, and incorrect access attempts waste time and produce cryptic error messages that provide no clue as to what the correct URL might be. A significant percentage of searches are for specific, well-known sites. These could be found much more quickly by a special-purpose locator engine. The current mode of interacting with search engines is also cumbersome-for this purpose, a much simplified mode of direct entry is practical.

One attempt to provide the ability to map a signifier, or alias, to a specific URL utilizes registration of

key words, or aliases, which when entered at a specified search engine, will associate the entered key word with the URL of the registered site. One such commercial implementation of this technique is known as NetWord (www.netword.com). However, the NetWord aliases are assigned on a registration basis, that is, owners of web sites pay NetWord a registration fee to be mapped to by a particular key word. As a result, the URL returned by NetWord may have little or no relation to what a user actually would be looking for. Another key word system, RealNames (www.realnames.com), similarly allows web site owners to register, for a fee, one or more "RealNames" that can be typed into browser incorporating RealNames' software, in lieu of a URL. Since RealNames also is registration based, there is no guarantee that the URL to which is user is directed will be the one he intended.

Further, in existing preference learning and rating mechanisms, such as collaborative filtering (CF) and relevance feedback (RF), the objective is to evaluate and rank the appeal of the best *n* out of *m* sites or pages or documents, where none of the *n* options are necessarily known to the user in advance, and no specific one is presumed to be intended. It is a matter of interest in any suitable hit, not intent for a specific target. Results may be evaluated in terms of precision (whether "poor" matches are included) and recall (whether "good" matches are not included).

A search for "IBM" may be for the IBM Web site, but it could just as likely be for articles about IBM as a

company, or articles with information on IBM-compatible PCs, etc. Typical searches are for information *about* the search term, and can be satisfied by any number of *"relevant"* items, any or all of which may be previously unknown to the searcher. In this sense there is no *specific* target object (page, document, record, etc.), only some open ended set of objects which may be useful with regard to the search term. The discovery search term does not signify a single intended object, but specifies a *term* (which is an attribute associated with one or more objects) presumed to lead to any number of relevant items. Expert searchers may use searches that specify the subject indirectly, to avoid spurious hits that happen to contain a more direct term. For example, searching for information about the book Gone With The Wind may be better done by searching for Margaret Mitchell, because the title will return too many irrelevant hits that are not about the book itself (but may be desired for some other task).

In other words, the general case of discovery searching that typical search engines are tuned to serve is one where a search is desired to return some number, n, of objects, all of which are relevant. A key performance metric, *recall*, is the completeness of the set of results returned. The case of a signifier for an object, is the special case of n=1. Only one specific item is sought. Items that are not intended are not desired--their relevance is zero, no matter how good or interesting they may be in another context. The top DirectHit for "Clinton" was a Monica Lewinsky page. That is probably not because people searching for

Clinton actually intended to get that page, but because of serendipity and temptation—which is a distraction, if what we want is to find the White House Web site.

In addition,

-CF obtains feedback from a group of users in order to serve each given user on an overall, non-contingent basis--without regard to the either the intent of the user at a specific time, or to being requested in a specific context.

-RF is used by a single user to provide feedback on their intent at a given time, but still with no presumed intent of a single target.

More broadly, searching techniques are generally not optimized based on using a descriptor which is also an *identifier*--they provide more generally for the descriptor to specify the nature of the content of the target, not its name. There are options in advanced search techniques which allow specification that the descriptor is actually an identifier, such as for searching by title. Such options may be used to constrain the search when a specific target happens to be intended, but no special provision is made to apply feedback to exploit that particular relationship or its *singularity*.

Moreover, none of the currently available key word systems utilize heuristic techniques actually to determine the site intended by the user. Instead, the current systems teach away from such an approach by their use of registration, rather than user intention,

to assign key words to map to web pages. Thus, the current techniques are not directed to solving the problem of finding the one, correct site for a particular signifier.

Thus, the need exists for a system that would enable a user to find a desired Web document by simply entering an intuitive key word or alias and that would perform a one to one mapping of the alias with the URL actually desired by the user, and which would use heuristic techniques to assist in providing the correct mapping, and improving system accuracy over time.

SUMMARY OF THE INVENTION

In consideration of the above deficiencies of the prior art, it is an object of the present invention to provide a method of signifier mapping that allows a user to locate to a particular network resource, in the preferred embodiment a web page, by simply entering a signifier or alias.

Thus, the present invention is generally directed to a technique for intelligent searching or matching where a *signifier* is given and is to be related to a name or address of an *intended* target object.

Signifier, in the context of the present invention means:

-an identifier, referent, or synonym for the name or address of a specific resource (a target object) presumed to exist in some domain; but

-not necessarily a "name" or "address"--a canonical identifier that has been assigned by some authority or pre-set by some convention (names are a subset of signifiers--those which are canonical or pre-established);

-not necessarily a description of content or subject matter (concepts or words);

-an identifier that has cognitive significance to the user, and presumed communication value in identifying the intended target object to another person or intelligent agent.

In addition, this cognitive/communication value is based on a perceived relationship (meant to have minimal ambiguity) to an identifier, which might be an assigned name or a name based on common usage, but which need not be exact, as long at it serves to signify the intended target.

More generally, descriptors may possibly be considered to be signifiers, if they are intended to be unique or minimally ambiguous (e.g. "the company that commercialized Mosaic" or "the company that sells the ThinkPad").

It is a further object of the present invention to provide a system in which heuristic techniques are used together with user feedback to improve the accuracy of signifier mapping.

None of the many solutions to the signifier mapping problem (Netword, Centraal, Goto, etc.) have identified

learning as a valuable technique. This may be because what naturally come to mind are techniques based on pre-defined mappings that make the use of "de jure" explicit registration. That teaches away from the idea of trying to learn the mappings heuristically from colloquial usage. (The same applies to attempts at creating systems for "user friendly names" in other directory systems.) Since the mappings are understood as being defined or registered, why would one try to learn about them? But actually, the mappings are just like natural language—they are dynamic, evolving, and ambiguous, and can only be resolved in terms of *learned usage within a context* — which is best addressed by learning, as in the present invention, not registration or other static mappings as appear in the prior art.

The use of heuristic, adaptive feedback-based techniques operates in significantly different ways when focused on signifier mapping, and this can be exploited by isolating such tasks. The key difference between the present invention and most common searching tasks is that in the prior searching techniques, there is no intention of a specific target object that is known to exist.

The present invention has several advantageous features, various combinations of which are possible:

1) a special purpose mapping engine for locating popular sites by guessed names;
2) automatic display of the target site (if located with reasonable confidence);

3) an optional simplified mode of direct entry of a guessed site name; and

4) use of user expectations, such as popularity of guesses intended for a given site, as a primary criterion for translating names to sites, with provision for protection of registered trademarks or other mandates.

In accordance with one aspect of the present invention, a finder or locator server is established. The server is configured to work with a user interface that allows users to enter an guessed name or alias, as easily as if the user knew the correct URL for the intended target resource. In response to entry of the alias, the finder server accesses a database that includes, in a preferred embodiment, conventional Web-crawler-derived index information, domain name registration information, as well as user feedback from previous users of the server, and looks up the correct URL, i.e., the one URL that corresponds to the alias and causes the user's browser to go automatically to that URL, without the user having to view and click on a search results page, if the correct URL can be determined with a predetermined degree of confidence.

In one preferred embodiment, the server is structured to accept the alias as a search argument and do a lookup of the correct URL and the return of same to the browser, without the intermediate step of the user having to wait for and then click on a search results web page. The automatic transfer is preferably effected using standard HTML facilities, such as a

redirect page or framing. Redirect is effected by placing pre-set redirection pages at the guess URL on the server. Alternately, the redirect page can be generated dynamically by program logic on the server that composes the page when requested.

The present invention advantageously uses feedback and heuristic techniques to improve the accuracy of the determination of the correct URL. If a suggested match is found by the look-up technique and the accuracy of the mapping is confirmed by user feedback, then, after directing the user to the URL, the result is stored in the server to improve the accuracy of subsequent queries. The server database includes a list of expected terms and expected variants that can initially be catalogued to provide for exact matches. This list is updated by the learning processes discussed in more detail below.

If, on the other hand, a probable one intended match cannot be determined, the finder server preferably uses intelligent techniques to find a selection of links to possible matches ranked in order of likelihood, or could return a no-match page with advice, or a conventional search interface or further directories.

According to a preferred embodiment of the invention, each of the selection of links are configured not to go directly to the target URL. Rather, the links are directed back to a redirect server established by the finder server, with coding that specifies the true

target, and feedback information. The finder server can in this way keep track of user selections.

In accordance with an advantageous aspect of the invention, such feedback information is used to improve the results of the search by promoting web sites almost universally selected to exact match status, and by improving the ranking of possible lists in accordance with which links are most often selected. Preferably, a confidence parameter can be generated from such tracking to control whether to redirect to a URL or to present a possible list to users.

In furtherance of the above and other objects, there is provided, a designated server, accessible on the Internet, the designated server being configured to respond to relocation requests that specify an identifier, corresponding to a target resource, that may not be directly resolvable by standard Internet Protocol name resolution services to the URL of the target resource. In a direct entry embodiment of the present invention, requests are passed to the relocation server by sending a relocation URL that designates the relocation server as the destination node and appends the identifying information for the identifier as part of a URL string. The relocation server extracts the identifying information and translates it into a valid URL for the target resource. The relocation server is configured, in the event that a unique URL can be determined with respect to the target resource, to cause the target resource to be

presented to the user without further action on the part of the user.

Preferably, the user requests are entered at a web browser, the relocation or search server determines the valid URL for the target resource by performing a look-up in a database, and the response from the relocation server is in the form of a redirect page that causes the user's web browser to obtain the target resource.

In accordance with one aspect of the present invention, there is provided a method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The method is for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained in the database. The method comprises: receiving a resource identity signifier from the user; and accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, the method further comprises directing a computer of the user so as to enable that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided an apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained in the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by: receiving a resource identity signifier from the user; and accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, a computer of the user is directed so as to cause that computer to connect the user to the address of the resource, if any, determined to be the intended target resource.

In accordance with yet another aspect of the present invention, there is provided a system for finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The system comprises: finder server means having: (a) database means for

storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system; and (b) learning system means for accessing and learning from information contained on the database; receiving means for receiving a resource identity signifier from the user; and accessing means for accessing the database means to determine which, if any, of the indexed resources is likely to be the desired target resource. Preferably, the system further comprises directing means for directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined in the access means to be the target resource.

In accordance with still another aspect of the present invention, there is provided a computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The method comprises: receiving a resource identity signifier from the user; and accessing the database to

determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, the method further comprises the step of: directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided a system for finding resources on a network of interconnected computers on which a plurality of resources reside. The system comprises: a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and a finder server having: (a) a database including: (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous operations of the system; and (b) a learning system operable to access and learn from information contained in the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by: receiving a resource identity signifier from the user; accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and directing a computer of the user so as to cause that computer to connect the user to the address of the

resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided a method of identifying, in response to entry by a user of an object identity signifier, a single intended object to be acted upon, the single intended object being intended by the user to uniquely correspond to the object identity signifier, among a plurality of possible objects. The method is for use on a computer having: (a) a database including (i) an index of possible objects; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained in the database. The method comprising: receiving an object identity signifier from the user; and accessing the database to determine, based upon the information in the database, which, if any, of the indexed objects is likely to be the object intended to be acted upon.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A is an architectural block diagram of a server computer system internetworked through the Internet in accordance with a preferred embodiment of the present invention;

Figure 1B is a flow diagram illustrating a method of obtaining feedback from multiple users to be applied in searching or signifier mapping;

Figure 2 is flow diagram showing a method of signifier mapping using feedback and heuristics to continually improve the performance of the mapping;

Figure 3 shows an example of a database entry for the finder server of the present invention;

Figure 4A is a flow diagram illustrating a technique of feedback weighting for probable results in signifier mapping;

Figure 4B is a flow diagram illustrating a technique of feedback weighting for possible results in signifier mapping; and

Figure 5 is a flow diagram illustrating how feedback is used in a preferred embodiment to discriminate a probable target resource in accordance with the present invention.

DETAILED DESCRIPTION OF THE DISCLOSED EMBODIMENTS

"Population cybernetics" and the Internet

As a general matter, the present invention relates to a technique that collects experience (a knowledge base) from a mass population that is open ended or universal, either over all domains, or over some definable subject or interest domain or strata. This represents a significant improvement over prior art techniques, which are generally limited in the scope of the

population and extent of experience from which they draw their knowledge base.

The technique of the present invention, in a preferred embodiment, uses the Internet to do this in a way that is powerful, economical, and far-reaching. The technique, in the preferred embodiment, uses the Internet to enable collection and maintenance of a far more complete knowledge base than has been used with any prior technique except Collaborative Filtering (CF).

In the present invention feedback learning is advantageously utilized, so that the information is not just collected, but refined based on feedback on the accuracy of prior inferences.

In its broad sense the present invention constitutes a kind of "population cybernetics," in that the learning does not just collect a linear knowledge base, but uses a feedback loop control process to amplify and converge it based on the results of prior inferences, and that it works over an entire population that is open, infinite, and inclusive. This is in contrast to prior learning techniques, which draw on necessarily finite, closed populations.

The use of population group information to achieve signifier mapping differs from the prior art technique of collaborative filtering in at least the following manner:

Whereas both CF and the technique of the present invention draw on knowledge of a population group to make inferences, CF obtains *ratings* of many things by many people to suggest other things (that may also be highly rated by the user, based on correlation with the group), and CF *does not involve* a *specific input request*, but rather seeks a *new, previously unknown item* in a category. On the other hand, the present invention obtains *translations* of many signifiers by many people to suggest the intended translation of a signifier and *involves a specific input request* to be translated to identify a *known intended target*

Although the technique of signifier mapping will occasionally be referred to loosely as searching, it is more accurately translation, because the target is intended and known, just not well specified. This differs from typical Web or document searching, which typically seeks unknown, new items.

The technique of the present invention also differs from natural language (NL) translation or understanding, in that the input is *atomic* in that it has no context as part of a body of discourse (a text). NL understanding techniques on the other hand translate words as components of concepts embedded in texts having a *context* of related ideas. Thus the cues of context in a discourse are absent, and the translation must be done without any such cues, although knowledge of the user may provide a useful context of behavior, demographics, psychographics that has some value in inferring intent, and knowledge of the user's prior

requests may provide additional useful context information. The task is to infer or predict intention, rather than to understand meaning, because there is no basis to infer meaning in any conceptual sense. The input is disjointed from any surrounding context, and if not seen before (from the user or others), there is little useful information on either its meaning or intention. The present invention seeks to infer intention based on limited data, primarily the input request, and draws on group data (of request translations) as its strength.

The task of the present invention has similarities with cryptanalysis, in that both the present invention and cryptanalysis use data about communications behavior from groups of communicators to make inferences. However the task differs in that

- Cryptanalysis deals with intentional hiding of meaning or intention, where the technique of the present invention is applied to cases where the hiding (of the intention of a signifier) is not at all intended; and
- Cryptanalysis seeks to infer meaning (ideas) drawing on context in a discourse, like NL understanding, not usually to infer the intention of a signifier (of objects or actions) which is not in a context.

This point of intention versus meaning is subtle, but has to do with communication of commands or requests as opposed to concepts.

- One view of this is the idea of "requests," as opposed to declarations or assertions, in the use of language.
- This task of recognizing commands (vs. meanings) has parallels in the task of robot control, such as that based on spoken commands. The similarity is in training understanding of the speech of many users to be speaker independent, and to infer meanings of a current speaker from that of others. The difference is that the tasks addressed in the present invention deal with a very wide, effectively infinite universe of commands (intended objects), while robot control techniques have generally been limited to very small sets of commands (partly because of the inability to apply mass experience).

Thus the technique of the present invention could be viewed as addressing a special class of robot control (in which experience data and feedback is accessible), and may ultimately be extensible to other robot control applications as such data becomes accessible over the network.

The social dimension is critical for inferences relating to shared objects or resources. Names draw on social conventions and shared usage. This social usage information is essential to effective mapping of signifiers to resources. De-jure naming systems can underlie a naming system, as for current Internet domain names, but de-facto usage is the essential observable source of information for fullest use. De-jure systems suffer from entropy, corruption and

substitution, while de-facto usage is pragmatic and convergent to changing usage patterns.

This applies to a variety of name-able resources:

- Web domain names;
- Web sub-site names (such as to find sub-areas);
- People or business names;
- Department, agent, or service identifiers (such as to to find contact points);
- Policy capability specifications (such as to find permissions, such as someone who can provide access to a given resource for a given purpose, such as confirming employment status or update-access to a report);
- Information sets or collections (to find reference tools that are known to exist, such as an IBM dictionary of acronyms, or an index of papers in ACM publications);
- Other robot control tasks, as social experience and feedback becomes accessible.

Social usage information can be combined with other sources of information in a heuristic fashion. For example, there could be a hierarchy that might be used in order, as available:

1. Personal defined usage information, such a defined personal nicknames;
2. Public de-jure defined mappings or directories;
3. Personal usage information (a person's own undefined nicknames, learned from that person's own usage/feedback);
4. Social de-facto usage information;

This is just one possible sequence, but shows how the usage data can take searching beyond what has been defined.

As discussed above, a preferred embodiment of the present invention relates to a method and apparatus for locating a desired target resource located and accessible on a network, in response to user entry of a guessed name or alias. In illustrating the preferred embodiment, the apparatus is shown as a server computer, or computers, located as a node on the Internet. However, the present invention is in no way limited to use on the Internet and will be useful on any network having addressable resources. Even more broadly, the present invention is useful for any similar task of identifying an intended target for an action in which automatic facilitation of that action is desired, where feedback from a large population can be obtained to learn whether a given response was in fact the one that was desired. Control of robots, as discussed herein, is one example of such broader application.

The finder server of the preferred embodiment of the present invention allows users to enter a guessed identifier or alias, as easily as if they knew the correct URL. Specifically, the finder server of the present invention accepts a guessed name, or alias, from a user, uses a look-up technique, enhanced by heuristics preferably taking into account previous users' actions, to determine a correct URL for the

intended target resource, and causes the user's browser to go to that URL automatically. Preferably this is done without the added step of first viewing and clicking on a search-results page, where an initial search finds the intended target resource with a predetermined degree of certainty. Such a resource will be referred to hereinafter as a "probable". In accordance with a preferred embodiment of the present invention, this functionality can be implemented by:

- Publicizing the locator server under an appropriate URL name, for example, guessfinder.com.
- Setting up the server to, in response to entry of a guessed name or alias, do a lookup to the correct URL and return a response that causes the user's browser to go automatically to the specified URL. Such an automatic transfer can be effected using a standard HTML facilities, such as a redirect page, or framing.
- If the guess does not provide an exact match in the lookup phase, using feedback and heuristic techniques to create and present to the user a selection of links to possible matches. Alternately, the user may be presented with a nomatch page with advice, or directed to a conventional search interface, or further directories.

It is contemplated that the use of aliases for attempting to locate a web site associated with company name or brand name would be found useful. For example, the aliases "s&p", "s-p", "sandp", "snp", "standardandpoors", "standardnpoors", "standardpoors" should preferably all map to www.standardpoors.com. In addition to companies and brands, other important name

domains would include publications, music groups, sports teams, and TV shows.

The present invention advantageously provides for learning and feedback on the basis of user preferences to automatically and dynamically build a directory of names and sites that maps to the actual expectations and intentions of a large population of users, and adapts to changes over time, including the appearance of new sites, thus optimizing utility to them.

The finder server of the present invention effectively provides a secondary name space, administered by the organization operating the finder system, through the automated heuristic methods described here, that maps to, but is not dependent on, the URL name space. The finder site computer has access to a data base containing entries for any number of popular sites, with any number of likely guesses and variations for each site.

As a result of the service provided when the present invention is implemented, site sponsors could skip the cumbersome and costly process of obtaining specific mnemonic URLs or alternate URLs in many cases (especially with regard to domain names). Even with a number of conventional URLS, this service could be a supplement, for additional variations. The problem of pre-empted URL domain names would also be avoided, except where there is legitimate and significant pre-existing usage.

A key to utility is to be able to directly connect in response to most guesses, and ambiguities could be a limiting factor. To avoid that it is desirable to exploit Pareto's Law/the 80-20 rule and do a direct connect even when there is an uncertain but likely target. For that to be useful, it must be easy for users to deal with false positives.

Correction after arrival at a wrong site can be made relatively painless by allowing a subsequent request to indicate an error in a way that ties to the prior request and adds information. For example a request, guessfinder.com/lionking, that located the movie but was meant to find the play could be corrected by entering guessfinder.com/lionking/play. A more efficient coding might explicitly indicate an error, such as guessfinder.com/!/lionking/play. Even with the error, this would be quicker and easier than conventional methods. Note that this example was illustrated with the direct URL coding techniques described below. Similar post-arrival corrections can be made with other user interface techniques, such as a frame header that includes appropriate user interface controls to report feedback, much as conventional search engines allow for "refinement" of prior searches, also described below.

Correction in-flight can be achieved by using the existing visibility of the redirect page, or enhancing it. When a redirect page is received by a user's browser, it appears for a short time (as specified with an HTML refresh parameter) while the target page is

being obtained. In addition to affording a way to optionally present revenue-generating (interstitial) advertising content, that page preferably lists the redirection target, as well as alternatives, allowing the user to see the resolution in time to interrupt it. This is most useful with a browser that permits a redirect to be stopped in mid-stream by clicking the stop button, leaving the redirect page on display, and allowing a correct selection among alternative links to be made. Alternately, a multi-frame (multi-pane) display could be used to allow a control frame to remain visible while the target page is loading in a results frame, as described below.

Note some of the typical parameters and control points that would be relevant:

"New" sites.
Applies when the user wants a site but is provided neither a direct hit, nor a correct possible. Users would find the site via alternate means (offered through the service or not). The user then submits an add-site request, via the Web or e-mail. If the number of add-site requests over a set interval exceeds a set (low) threshold, the site is added as a possible, or a direct hit if there are no competing alternatives. Such adds would be provisional, and could be dropped if requests are not sustained.

Possibles

Low confidence possibles would be listed low on the list, and selections would be tracked. If selections are strong, they move up the list. If selections are very weak, they would drop off after some interval. The threshold to add back sites that were dropped might be higher for a time, to limit oscillation and false adds. If possibles are well ahead of alternatives by some threshold over some interval, they would be promoted to direct hits.

Direct hits

Feedback on false positives would be collected. This could be via links in frames, redirect pages, interstitials, or other means, as suggested previously. If false positives exceed a threshold, the site would revert to a possible and the common alternatives would be listed as well.

Parameter issues: thresholds, intervals, smoothing, damping, overrides.

Basic parameters include the various thresholds and time intervals for measurement. Smoothing techniques (such as exponential smoothing) would be applied to adjust for random variations and spikes, to improve forecasting. Damping mechanisms could be used to limit undue oscillation from state to state. Overrides would provide for mandated or priority matches, such as for registered trademarks, on either a weighted or absolute basis, as appropriate.

Figure 1A illustrates a first embodiment of the present invention, as implemented on the Internet. The finder server 10 includes a computer or computers that perform processing, communication, and data storage to implement the finder service. Finder server 10 includes a finder processing/learning module 101. Module 101 performs various processing functions, and includes a communication interface to transmit and receive to and from the Internet 12, as well as with database 102, and is programmed to be operable to learn from experiential feedback data by executing heuristic algorithms. Database 102 stores, in a preferred embodiment, indexes of URL data that would allow the module 101 to locate, with a high degree of confidence, a URL on the Web that is an exact match for a target resource in response to a user's entry of an alias or guessed name. Preferably, the indexes store, in addition to available URL information, such as domain name directories, information relating to the experience of the server in previous executions of the finder service. As the server gains experience and user feedback, heuristic techniques are applied by module 101 to enable the returned URLs to conform more and more accurately to user expectations.

Users 11_0-11_N can access the Internet 12 by means of client computers (not shown) either directly or through an Internet service provider (ISP). As has been discussed previously, to make use of the present invention the user enters a guessed name, or alias, into his computer's browser and submits a query containing the alias to the finder server. The World

Wide Web 14 includes computers supporting HTTP protocol connected to the Internet, each computer having associated therewith one or more URLs, each of which forming the address of a target resource. Other Internet information sources, including FTP, Gopher and other static information sources are not shown in the figure.

The finder server includes operating system servers for external communications with the Internet and with resources accessible over the Internet. Although the present invention is particularly useful in mapping to Internet resources, as was discussed above, the method and apparatus of the present invention can be utilized with any network having distributed resources.

Entry of the alias by a user may be accomplished in a number of ways. In one embodiment, a usage convention can be publicized for passing the alias to the server within a URL string, such as guessfinder.com/get?ibm, for example, for trying to find the web page corresponding to the alias "ibm". In this case, the server is programmed to treat the string "ibm" as a search argument and perform the appropriate processing to map the alias to the intended target resource. A similar effect can be obtained by the somewhat simpler form guessfinder.com/ibm, if the server is programmed appropriately. Alternately, the user can visit the web site of the finder server and be presented with a search form, as is done in conventional search engines. A third option is to provide a browser plug in that allows direct entry of the key word in the browser's

URL window or any alternative local user interface control that will then pass the entry on as a suitably formatted HTTP request.

It also would be preferable for an enhanced user interface to be phased in as the service gains popularity. This preferably would be accomplished by a browser plug-in, or modifications to the browser itself, to allow the alias to be typed into the URL entry box without need for the service domain name prefix (such as, guessfinder.com/...). Instead, such an entry would be recognized as a alias, not a URL, and the prefix would be appended automatically, just as http://... is appended if not entered with a URL in current browsers.

Figure 1B is a flow diagram illustrating a technique for obtaining and learning from feedback responses gathered from a large group of people, in the example, users 1, 2, . . . n. Such a technique can be used in a variety of applications, and in particular in traditional search engines, or in mapping to identify particular web sites, as in alias or signifier mapping.

In Figure 1B, users 1, 2, . . . n represent a large community of users. In the flow diagram, the flow of query items from the users is indicated by a Q, the flow of responses back to the users is indicated by an R, and the flow of feedback results provided by the users' actions, or responses to inquiries, is indicated by an F. As can be seen from the figure, Query (a, 1) is transmitted from user 1 to the service 2, which can

either be a searching or a mapping service. The service has learning processor 4, which interfaces with a database 6. The database 6 contains, among other things, indexes and feedback information gathered from previous queries. In response to the query, the user 1 is provided with a response R(a, 1). User 1 then is provided with the opportunity to transmit user Feedback (a, 1) to the Service 2. Learning processor 4 stores the feedback information in the database 6, and is programmed with one or more heuristic algorithms enabling it to learn from the feedback information to improve the returned search or mapping results. The feedback provided will improve the results offered, for example by positively weighting results preferred by users, so that, over time, more accurate results can be obtained.

Figure 2 is a diagram illustrating the logical flow used in applying the general technique of learning from user feedback shown in Figure 1 to signifier mapping, in accordance with a preferred embodiment of the present invention. A user enters a Query consisting of a signifier, represented by Q^s. The server, in response to receipt of the query, parses the query, at step S02, and in step S04 performs a database lookup in an attempt to determine, if possible, the exact target resource intended by the user. Database 6 includes index data as well as feedback data obtained from users in previous iterations of the signifier mapping program, is accessed. The stored data structure is described in more detail below.

In step S06, the program discriminates a probable intended target making use of the index information such as domain registration indexes, and other resources, as well as the feedback information stored in the database. In step S08, if a likely hit, or exact match has been identified, that is, a web page has been located with a high confidence parameter, the flow continues to step S10. At step S10, a direction is prepared to the likely hit URL. A list of alternatives optionally may be provided for presentation to the user at the same time, in case the likely hit turns out not to be the target identifier. At step S12, the server sends information R^s to the user, more particularly to the user's browser, to effect a link to the likely hit. Optionally, the alternate list is also provided at the same time.

In step S14, the viewed page is monitored by the server and the user, by his actions, provides feedback. Most readily determined with no assistance from the user is the fact of the user having chosen the link. This may be determined, for example by a redirect, in which an intermediate server is transparently interposed between the browser and the target page, and thus able to identify the user and the URL target based on coding built into the URL that the user clicks. Also desirable is the amount of time the user spends at the site, which will be an indicator of whether the site is the intended target. This may be ascertained, for example, if clickstream data can be obtained, such as through the use of a monitor program that works as a browser add-in or Web accessory, such as the techniques

offered by Alexa. Other feedback can be provided by asking the user. This can, for example, be done conveniently by using a small header frame served by the relocation service that appears above the actual target page, and that includes controls for the user to indicate whether or not the results were correct. The URL of the viewed page is recorded, together with any other feedback, for use in improving the accuracy of subsequent iterations of signifier mapping. At step S26, the feedback data is supplied to a feedback weighting algorithm, described in detail below, which generates appropriate weighting factors to be stored in the database for use in subsequent mappings.

If it is determined at step S08 that the result is not a likely hit, the flow proceeds to step S18, where a list of the top m hits (m being a predetermined cutoff number), preferably drawing on the list of possible hits from a conventional search engine, or by employing the same techniques as a conventional search engine, is prepared. Unlike conventional search engines, the ranking of these hits is based primarily on experience feedback data as described below. In addition, where such feedback is limited or absent, it would be supplemented by variants of more conventional search engine weighting rules that are expressly tuned to the task of finding a single intended result (i.e., high relevance by low recall) rather than many results (high relevance plus high recall). The list is presented, at step S20, to the user as R^s. The user, by the selections made from the provided list, and from other feedback, such as how long the user spends at each

link, supplies feedback to the system. This information F^s is monitored, at step S22 and recorded, at step S24. The recorded information is supplied to the feedback weighting algorithm, at step S26, the output of which is stored in the database for use in subsequent iterations of the signifier mapping.

Figure 2 illustrates the simple case in which a user is directed to a target URL if the target has been determined to be a probable hit, and is presented with a list to choose from if the target cannot be identified with sufficient certainty. However, it is well within the intended scope of the invention for alternate methods to be employed. For example, the user interface (UI) could be extended, either by framing, or a browser plug-in or extension, to provide multi-pane/multi-window results that allow a pane for each type of response, e.g., the target response and a list of possibles, regardless of the level of confidence in the result. In such a case, the format for presentation of results would be the same whether a probable has been located or not, but the learning from feedback and ranking would still seek to determines "correctness" based on the varying feedback cases.

Figure 3 illustrates a preferred method of organizing index data to allow for storing and updating of the most probable hits for a given query. As can be seen from the illustration, for each query, whether single element queries or compound queries, there is stored a list of associated possible targets. Linked to each of these query/target pairs is a raw score, an experience

level, and a probability factor. As feedback enters the system, the index data is updated to reflect the user feedback. The updating process will be described below. While the index shows preferred weighting criteria, these are only a sample of the kind of criteria that can be correlated to the query/target pairs. In a simple embodiment, the raw score would be based only on selections of hits, and explicit feedback on correctness as described below. Other embodiments could add feedback data on time spent at a target. Additional variations would include weighting based on the recency of the feedback, and on the inclusion of non-feedback data, such as the various syntactic and semantic criteria used for relevance weighting by conventional search engines.

The process of maintaining the guess-target database is adaptable to a high degree of automation, and this can be highly responsive to new sites. An outline of such a method is:

All guesses are logged and analyzed.

Ambiguous hits are tracked as described earlier.

Complete non-matches are sorted by frequency to identify common new requests (in real time). Changes in ambiguous match patterns could also flag appearance of new sites.

Common new requests preferably are fed to an automated search tool that would use existing search engines, hot site lists, and name registration servers, etc. to identify possible targets.

Automated intelligent analysis of those results can seek to qualify probable targets.

High confidence (or possible) targets preferably are added, and then tracked based on the feedback mechanism described earlier, in order to self-correct. A confidence parameter preferably is used to control whether to redirect or to present a possibles list to users.

Human review and correction also preferably is used to supplement this.

Figure 4A illustrates a preferred technique for weighting the results using feedback data for hits that have been determined to be probable hits. In step S30, if the user feedback from the probable result indicates that the probable was in fact the target URL the user was searching for, the flow proceeds to step S32 where the raw score for that query/target pair is incremented by $factor_Y$. If the user returns feedback indicating that the probable was not the target resource the user had in mind, the flow proceeds to step S34 where the raw score for that query/target pair is decremented by $factor_N$. If the user provides no feedback, then the flow proceeds to step S36 where the raw score is decremented by $factor_O$, which can be zero. After execution of any of steps S32, S34 or S36, the flow proceeds to step S38, at which the experience level score is incremented by $Efactor_C$.

Figure 4B illustrates a preferred technique for weighting in accordance with user feedback in the case of possibles, i.e., items on the list presented to the user when no probable result can be located. As shown in the figure, if a possible is selected by the user

from the presented list, at step S40, the fact of selection is recognized, preferably by use of a redirect server that allows the system to keep track of which link was chosen. Additionally, the amount of time the user spends at the selected link may be ascertained. Making use of the information gathered in the redirect and such other feedback as may be obtained, the raw score for the query/target pair is incremented, at step S44, by $factor_S$. The user is then requested to provide additional feedback after the user has finished viewing the link.

In a preferred embodiment of the present invention, this feedback is gathered from the user by presenting the user with a frame that includes a mechanism, such as a check box, or radio button, that allows the user to indicate whether the selected possible was in fact the intended or "correct" target resource. If it is determined, at step S42, from the feedback that the link was the correct target, the flow proceeds to step S46, where the raw score for that query/target pair is incremented by $factor_{Y'}$. If the user returns a negative response, the raw score of the pair is decremented at step S48 by a by $factor_{N'}$. If no feedback is received, the raw score is decremented, at step S50, by $factor_{O'}$, which can be zero. After execution of any of steps S44, S46, S48 or S50, the flow proceeds to step S32, at which the experience level score is incremented by $Efactor_{PS}$ in the case of selection of the link, and by $Efactor_{PC}$ if the link was the correct.

Figure 5 illustrates a detail of how the present invention ranks and discriminates a probable target. At step S100 a list of possibles is obtained. Next, the list is ranked, at step S102, on the basis of the expected probability as the target. In step S104, a discrimination criteria is calculated and compared with a predetermined threshold parameter. For example, if ProbTi is the expected probability that Ti is the correct target, a formula such as the example shown can be used to determine whether T1 stands out as more probable than T2 by a relative margin that exceeds a set threshold needed to judge it as the probable intended one target. When the threshold is not exceeded, the implication is that one of the secondary possibilities may very well be the intended one, and that directing the user to the slightly favored target may not be desirable.

In the preferred embodiment, when a link on a list of possibles is selected by the user, rather than connect the user immediately to the chosen link, the finder server first redirects the user to a redirect server where feedback data relating to the selection can be gathered. One item of feedback that may be obtained in this manner is the very fact of the selection. Further feedback can be obtained by additional means, such as monitoring how long the user spends at the selected link, and by directly querying the user.

The redirect linking technique uses the target URL as a server parameter within a composite URL to control the intermediate server parameter within the URL to control

the intermediate server. The target URL is embedded as a server parameter within a URL that addresses the redirect server, and the URL parameter is used to control the intermediate server process. Thus a server is called with a first URL, a redirect URL, that specifies the second URL, i.e., the target URL, as a parameter. For example

http://redirector.com/redirector?query12345678/targetserver.com/targetpath1/targetpage1.htm

where redirector.com is the intermediate server URL, query12345678 is a unique identifier of the user-query combination, and targetserver.com/targetpath1/targetpage1.htm is the target URL. The network ignores the parameter portion of the URL, which is passed as data to the server. The server acts on the parameter to perform desired intermediary processing, in this case, the logging of the fact that this link was clicked in response to query12345678, and to redirect the user to the intended location specified by the second URL. The token query12345678 could be a unique identifier corresponding to a logged user-query entry, or it could be the actual query string.

The delay required for the redirect provides the opportunity for the display of interstitial advertisements. In addition, additional user feedback can be solicited during the delay, and the connection to the targeted URL can be aborted if the user indicates that the target site is not the one he or she

intended. In addition to using the redirect when a link is selected, the technique also preferably is used when an exact match is found, to provide a brief delay before connecting the user to the exact match, to present advertisements to give the user the time to abort the connection. In any event, the user preferably is given the opportunity to provide feedback after connecting to any site, whether directly as a result of an exact match, or as a result of selecting from a linked possibles list.

The redirect server of the present invention allows data to be gathered on each link as it is followed and redirected. The redirect link can be created in a simple static HTML. However, it is preferable to create the link dynamically for each user selection.

The finder is setup to recognize the feedback function, possibly as a CGI or other gateway/API function, and invoke the appropriate function to parse the URL or other data (referer, cookies, etc.), extract the target URL and feedback information for processing, and return a page containing a redirect (or use framing or other means) to take the user to the desired target.

This mechanism is general, and can be used for many purposes. In the case of the finder server:

Reasonably complex feedback information can be obtained, which at minimum would include the original guess. Thus a log of each guess that was not clearly

resolved, paired with the corresponding user-selected target, can be obtained.

-That set of selected guess/target pairs can then be used to adjust the confidence levels in the guess/target database. Similar data on directly resolved pairs would also be applied, along with any data from wrong-match reports.

Other applications are to any situation where links go to sites other than the source. This would include results of conventional search engines, as well as resource directories, sites referring users to suppliers, advertisers, etc.

The above embodiments of the present invention have been described for purposes of illustrating how the invention may be made and used. The examples are relatively simple illustrations of the general nature of the many possible algorithms for applying feedback data that are possible. However, it should be understood that the present invention is not limited to the illustrated embodiments and that other variations and modifications of the invention and its various aspects will become apparent, after having read this disclosure, to those skilled in the art, all such variations and modifications being contemplated as falling within the scope of the invention, which is defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained in the database, the method comprising:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

2. A method according to Claim 1, further comprising:

directing a computer of the user so as to enable that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

3. A method according to Claim 1, wherein a resource is determined, at the accessing step, as likely to be the intended target resource if the database information indicates that a confidence level

associated with that resource is of at least a predetermined level.

4. A method according to Claim 3, wherein if none of the indexed resources have an associated confidence level of at least the predetermined level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

5. A method according to Claim 3, wherein the method further comprises the following steps:

in a first window of the user's computer screen:

causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second window of the user's computer screen:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

6. A method according to Claim 4, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, incrementing the confidence factor associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

7. A method according to Claim 2, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, incrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

8. An apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered in previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained in the database;

the finder server being operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

9. An apparatus according to Claim 8, wherein the finder server is further operable to:

direct a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined to be the intended target resource.

10. An apparatus according to Claim 8, wherein a resource is determined, in the accessing, to be the intended target resource if the database information

indicates that a confidence level associated with that resource is of at least a predetermined level.

11. An apparatus according to Claim 10, wherein the apparatus is operable to, if none of the indexed resources have an associated confidence level of at least the predetermined level, perform the following steps:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest;

request that the user select a link; and

cause the user's computer to go to the selected link, if any.

12. An apparatus according to Claim 10, wherein the apparatus is further operable to:

in a first window of the user's computer screen:

cause the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second window of the user's computer screen:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

13. An apparatus according to Claim 11, wherein the apparatus is operable to, if a link has been selected, perform the following steps:

add information regarding the selection of the link to the feedback information in the database;

solicit user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, increment the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, decrement the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

14. An apparatus according to Claim 9, the apparatus being further operable to:

solicit user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, increment the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, decrement the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

15. A system for finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the system comprising:

finder server means having: (a) database means for storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system; and (b) learning system means for accessing and learning from information contained on the database;

receiving means for receiving a resource identity signifier from the user; and

accessing means for accessing the database means to determine which, if any, of the indexed resources is likely to be the desired target resource.

16. A system according to Claim 15, further comprising:

directing means for directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined in the access means to be the target resource.

17. A system according to Claim 15, wherein a resource is determined, by the access means, as likely to be the intended target resource if the database information indicates that a confidence level associated with that resource is of at least a predetermined level.

18. A system according to Claim 17, further comprising:

presenting means for, if none of the indexed resources have an associated confidence level of at least the predetermined level, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest;

requesting means for requesting that the user select a link from the presented list; and

means for causing the user's computer to go to the selected link, if any, upon a user selection of a link.

19. A system according to Claim 17, further comprising:

means for, in a first window of the user's computer screen, causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

means for, in a second window of the user's computer screen, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest

20. A system according to Claim 18, further comprising:

adding means for, if a link has been selected, adding information regarding the selection of the link to the feedback information in the database;

soliciting means for soliciting user feedback with regard to the selected link; and

means for, if the user indicates that the link is the resource intended by the resource identity signifier, incrementing the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

21. A system according to Claim 16, further comprising:

soliciting means for soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

means for, if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, incrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

22. A computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method comprising:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

23. A computer-readable medium according to Claim 22, wherein the method further comprises the step of:

directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

24. A computer-readable medium according to Claim 22, wherein a resource is determined, in the accessing step, as likely to be the intended target resource if the database information indicates that a confidence

level associated with that resource is of at least a predetermined level.

25. A computer-readable medium according to Claim 24, wherein if none of the indexed resources have an associated confidence level of at least the predetermined level, the method further comprises the following steps:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest;

requesting that the user select a link; and

causing the user's computer to go to the selected link, if any.

26. A computer-readable medium according to Claim 24, wherein the method further comprises the following steps:

in a first window of the user's computer screen:

causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second window of the user's computer screen:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

27. A computer-readable medium according to Claim 25, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, incrementing the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

28. A computer-readable medium according to Claim 23, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, incrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, decrementing the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

29. A system for finding resources on a network of interconnected computers on which a plurality of resources reside, the system comprising:

a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and

a finder server having:

(a) a database including: (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous operations of the system; and

(b) a learning system operable to access and learn from information contained in the database,

the finder server being operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by:

receiving a resource identity signifier from the user;

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and

directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

30. A method of identifying, in response to entry by a user of an object identity signifier, a single intended

object to be acted upon, the single intended object being intended by the user to uniquely correspond to the object identity signifier, among a plurality of possible objects, the method for use on a computer having: (a) a database including (i) an index of possible objects; and (ii) information regarding user feedback gathered in previous executions of the method; and (b) a learning system structured to access and learn from information contained in the database, the method comprising:

receiving an object identity signifier from the user; and

accessing the database to determine, based upon the information in the database, which, if any, of the indexed objects is likely to be the object intended to be acted upon.

ABSTRACT OF THE DISCLOSURE

An apparatus for finding resources on a network comprises a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered from previous operations of the apparatus; and (b) a learning system operable to access and learn from information contained on the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by: receiving a resource identity signifier from the user; accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

Date: 3/20/00
Sender: rreisman@rxremedy.com (Richard Reisman)
To: Joseph W Ragusa
cc: bgb@btgusa.com
Priority: Normal
Subject: Comments on draft - CUL "final"

REDACTED

--

Richard Reisman - RxRemedy/HealthSCOUT - Operations & Technology, CTO
52 Vanderbilt Ave., 9 Fl., NY, NY 10017 - (646)-227-7622 (fax 227-7601)
http://www.healthscout.com e-mail: rreisman@rxremedy.com
[Teleshuttle Corp. - http://www.teleshuttle.com - rreisman@teleshuttle.com]
[MIT Enterprise Forum of NYC - Digital Media Program Chair]

===

CUL-RR-3-19.zip

RR Comments on 3/16 "Final" revision.

Disclosure:

1. Page 42, line 27, change "proceeds to step S32" to "...S52"

2. Adapt and include additional material on robot control from the task domain filing (after p46, line 14?):

   While the present invention has been discussed primarily in terms of its applicability to searching the Web, the concept has much broader applicability. For example, in the area of robot control, the above techniques can be used to allow a robot to understand more readily the actual intent of a command.

   For example, in the general case, analogous to discovery searching, the robot command may be performable in many ways, such as "direct the excess inventory out of the active holding bin," allowing the robot to find any of several allowed places to move the inventory to, and leaving some degree of ambiguity that complicates translation. In the n=1 case, or signifier mapping, more specific feedback heuristics can be utilized as ~~described above for Web signifier searches, to assist the robot in~~ determining the one acceptable action to be taken in response to the command such as "direct the excess inventory to the secondary holding bin."

   Also some material from my 1/16/00 fax, p2 may be useful here.

   Note my question on how much duplication we want across the two filings: What issues re including more concept background from CUL, such as population cybernetics ideas, feedback details (redirect), etc.
   - Does this better enable claims elaboration?
   - Does this complicate separate sale of patents by muddying which includes what elements?

Claims:

1. "by a multiplicity of users" I thought we had agreed that dependent claims were to be added as suggested in my prior comments to include the idea that the feedback came from many users. This is probably essential to distinguish from some prior art, which may use learning from one or a few users.

2. A corresponding change to the abstract was also suggested in my prior comments.

3. Claims 4, 11, 18, 25: I think we can drop the last two clauses ("requesting that the user select a link; and causing the user's computer to go to the selected link, if any")as being implied, non-essential, and better left unstated.

4. Claims 5, etc.: Suggest changing "window of the user's computer screen" to "user interface element" as being more general and future-proof.

5. Claims 7, etc.: I think we can do better by changing to "...**updating the database information so as to increase [/ decrease]** the confidence level...." This states the essence without limiting the details of the method--eg: what part is done at DB update time and what part is done as search time.

   Note: I was also concerned that "level" could be limiting/confusing in this specific use and that "factor" seemed to be a better word than "level" here, since it is a factor used to determine a confidence level. "Data" may even be a better, more general word. My leaning would have been to use data here, but I think the above suggestion is preferable.
   Note "level" is a good word as now used elsewhere regarding the outcome of the decision process, as in Claim 3, and that change from "factor" to "level" is what I intended, and should stand.)

Diagrams

1. Fig. 2: Add close quote in S10.

2. Fig 4B: S48 should point to the box Decrement raw score by factorN.

File

FITZPATRICK, CELLA, HARPER & SCINTO

30 ROCKEFELLER PLAZA
NEW YORK, NY 10112-3801
212-218-2100
FACSIMILE (212) 218-2200

WASHINGTON OFFICE
1900 K STREET, N.W.
WASHINGTON, D.C. 20006-1110
(202) 530-1010
FACSIMILE (202) 530-1055

CALIFORNIA OFFICE
650 TOWN CENTER DRIVE, SUITE 1800
COSTA MESA, CALIFORNIA 92626-1925
(714) 540-8700
FACSIMILE (714) 540-9823

WRITER'S DIRECT DIAL NUMBER
(212) 218-2223

March 31, 2000

VIA FEDERAL EXPRESS

Bruce C. Bernstein, Esq.
BTG International Inc.
2200 Renaissance Blvd.
Gulph Mills, PA 19406

Re: Final Draft of CUL Application
Our Ref.: 1311.1100

Dear Bruce:

I enclose a copy of a patent application directed to Dick Reisman's CUL invention.

REDACTED

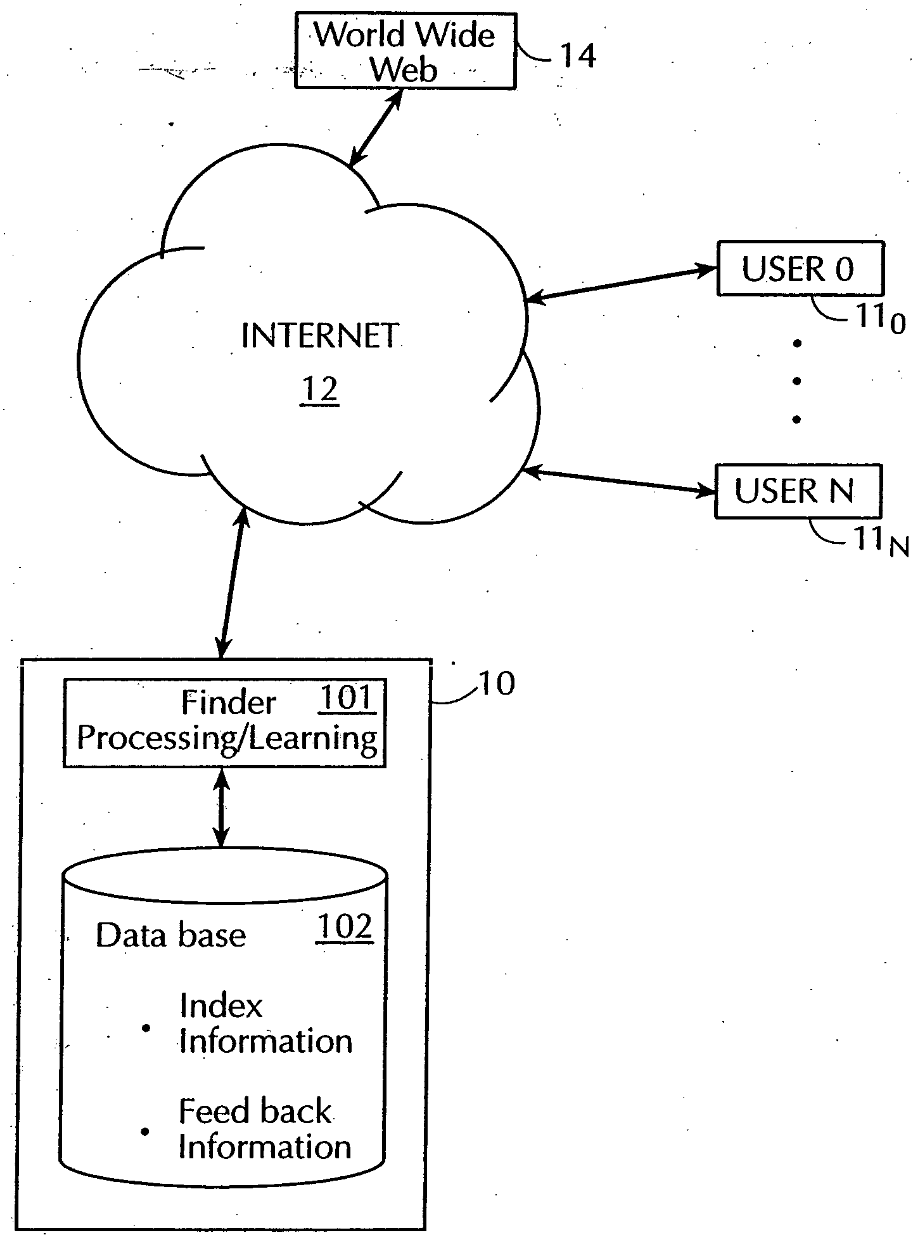
Sincerely yours,

Joseph W. Ragusa

Joseph W. Ragusa

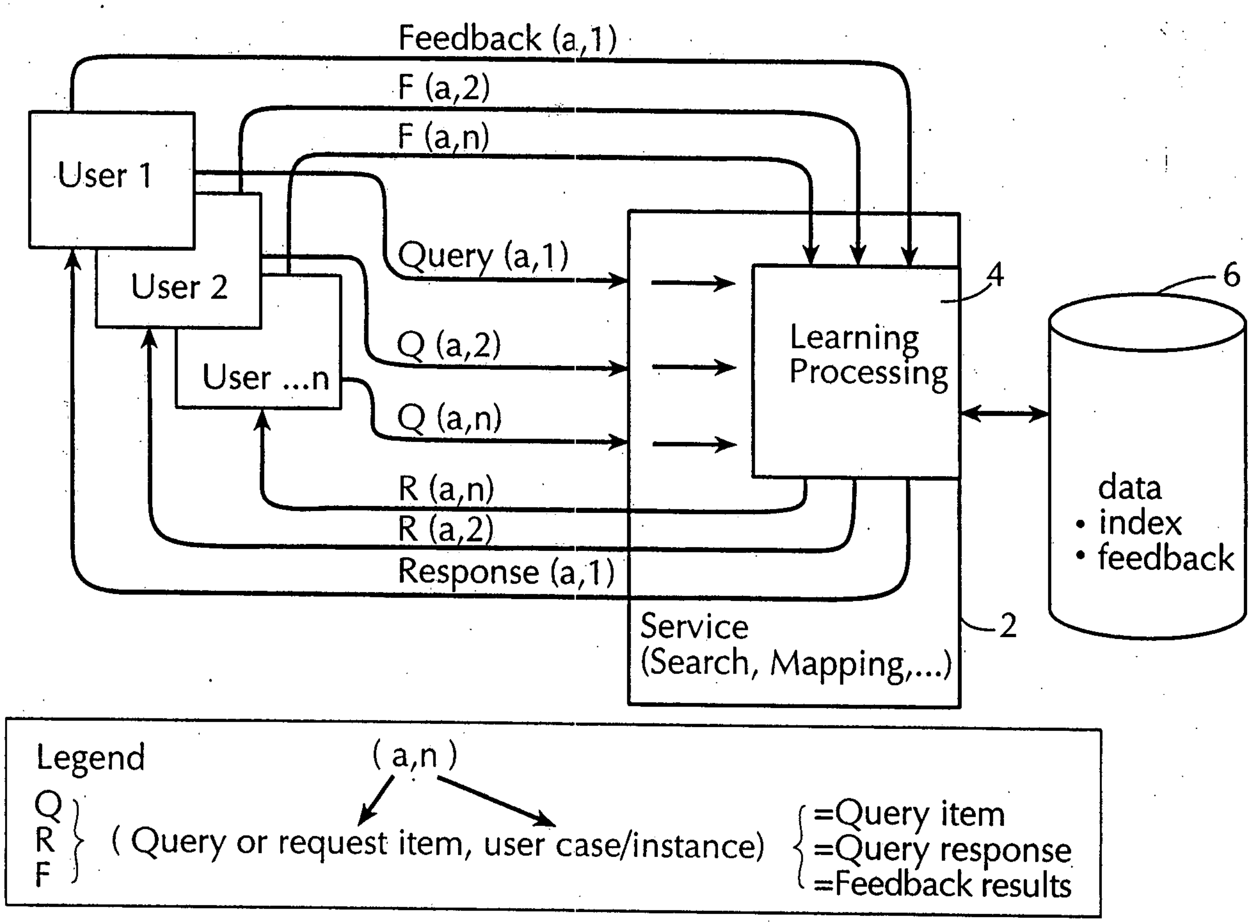
Enclosure
cc: Dick Reisman (w/enclosure)

FIG. 1A



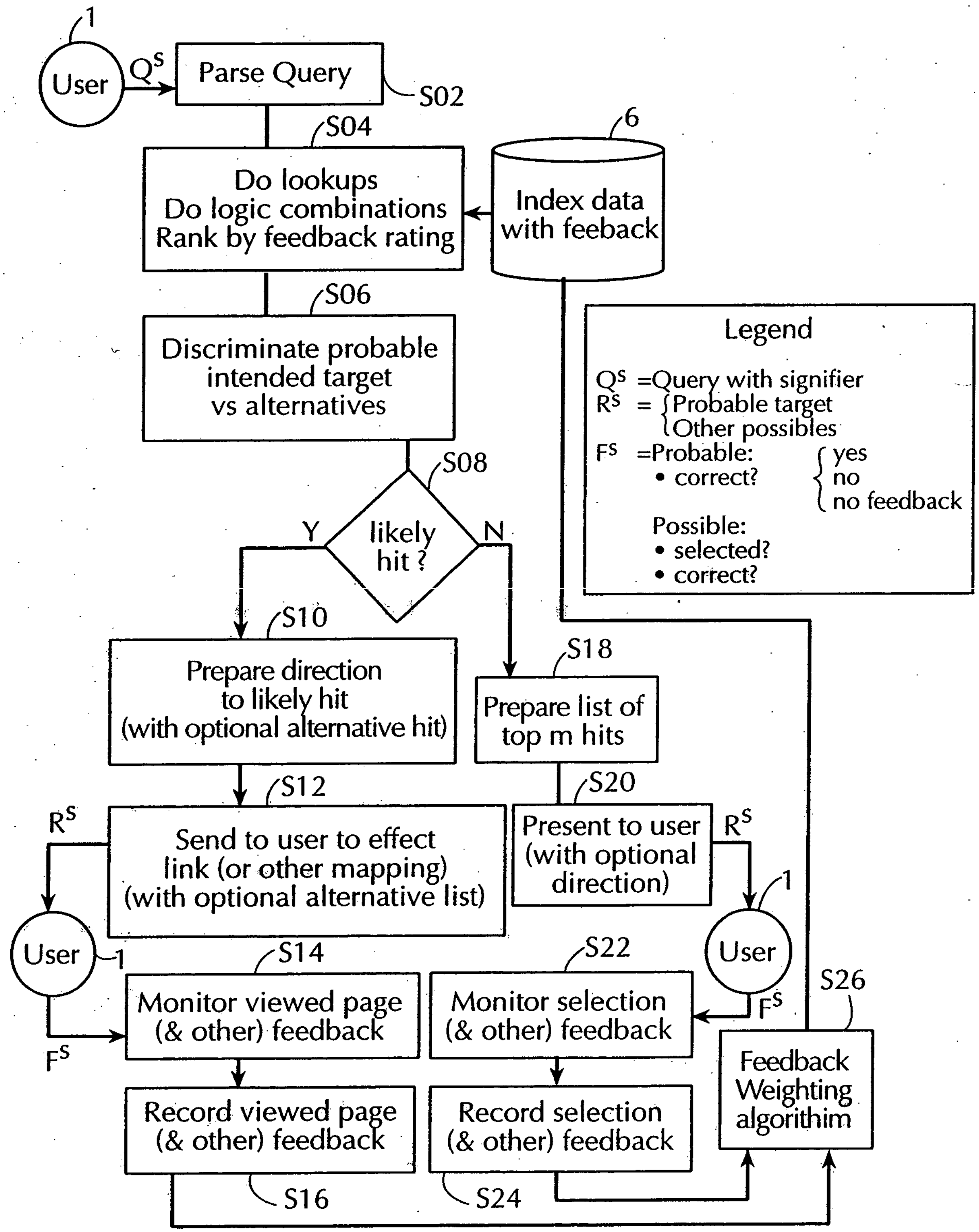
World Wide Web
14
INTERNET
12
USER 0
11_0
USER N
11_N
10
Finder 101
Processing/Learning
Data base 102
Index Information
Feed back Information

FIG. 1B



Feedback (a,1)
F (a,2)
F (a,n)
User 1
User 2
User ...n
Query (a,1)
Q (a,2)
Q (a,n)
R (a,n)
R (a,2)
Response (a,1)
Learning
Processing
4
6
data
• index
• feedback
Service
(Search, Mapping,...)
2
Legend
(a,n)
Q
R
F
(Query or request item, user case/instance)
=Query item
=Query response
=Feedback results

FIG. 2



1
User
Q^S
Parse Query
S02
S04
Do lookups
Do logic combinations
Rank by feedback rating
6
Index data
with feeback
S06
Discriminate probable
intended target
vs alternatives
S08
likely
hit ?
Y
N
Legend
Q^S =Query with signifier
R^S = Probable target
Other possibles
F^S =Probable:
• correct?
yes
no
no feedback
Possible:
• selected?
• correct?
S10
Prepare direction
to likely hit
(with optional alternative hit)
S18
Prepare list of
top m hits
S12
R^S
Send to user to effect
link (or other mapping)
(with optional alternative list)
S20
Present to user
(with optional
direction)
R^S
1
User
1
User
S14
Monitor viewed page
(& other) feedback
S22
Monitor selection
(& other) feedback
F^S
F^S
S26
Feedback
Weighting
algorithim
Record viewed page
(& other) feedback
Record selection
(& other) feedback
S16
S24

FIG. 3

Single element Q's	Possible Targets	Raw score	Experience level	Probability factor
Q (a)	1	-	-	-
	2	-	-	-
	3	-	-	-
Q (b)	1	-	-	-
	⋮	⋮	⋮	⋮
Compound/ (multi-element) Q's				
Q (c)	1	-	-	-
	2	-	-	-
	⋮	⋮	⋮	⋮

FIG. 4A

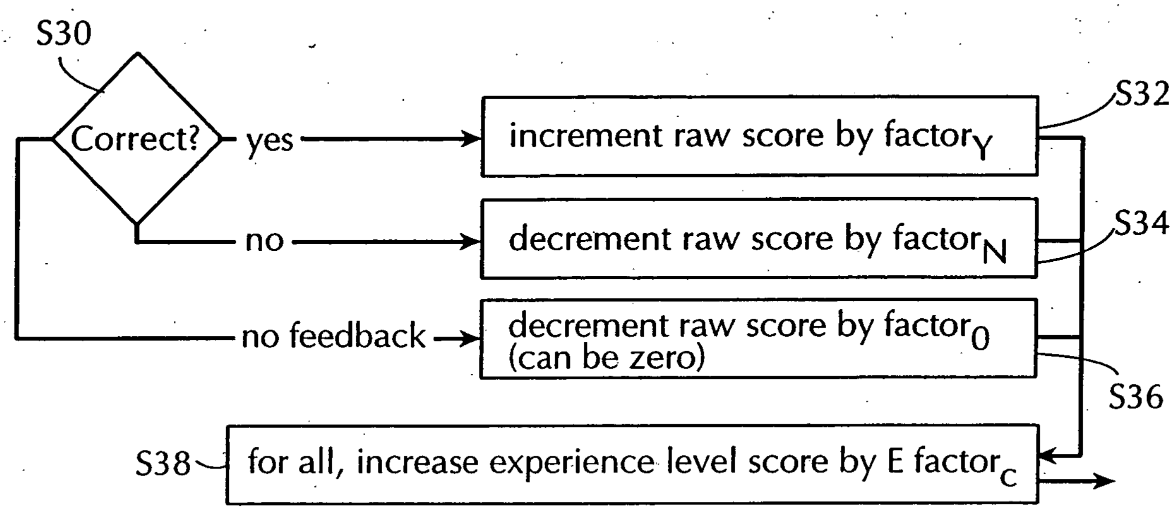

S30
Correct?
yes
no
no feedback
increment raw score by factor$_Y$
S32
decrement raw score by factor$_N$
S34
decrement raw score by factor$_0$
(can be zero)
S36
S38
for all, increase experience level score by E factor$_c$

FIG. 4B

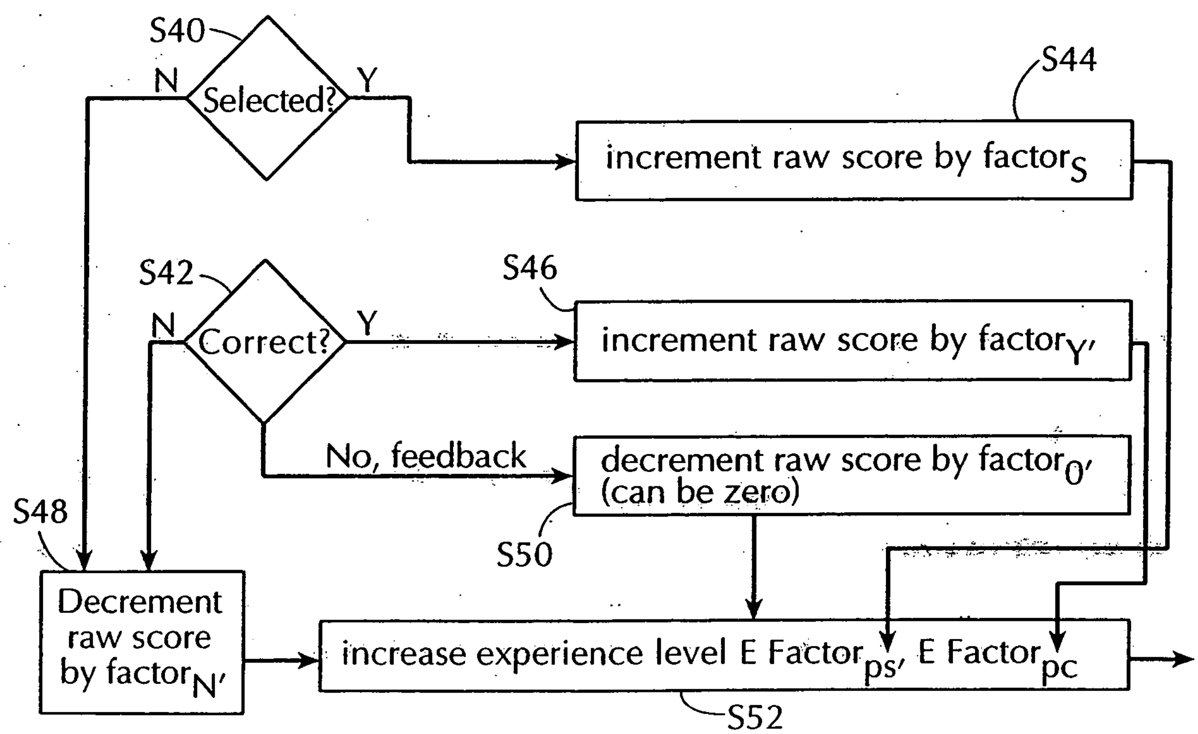
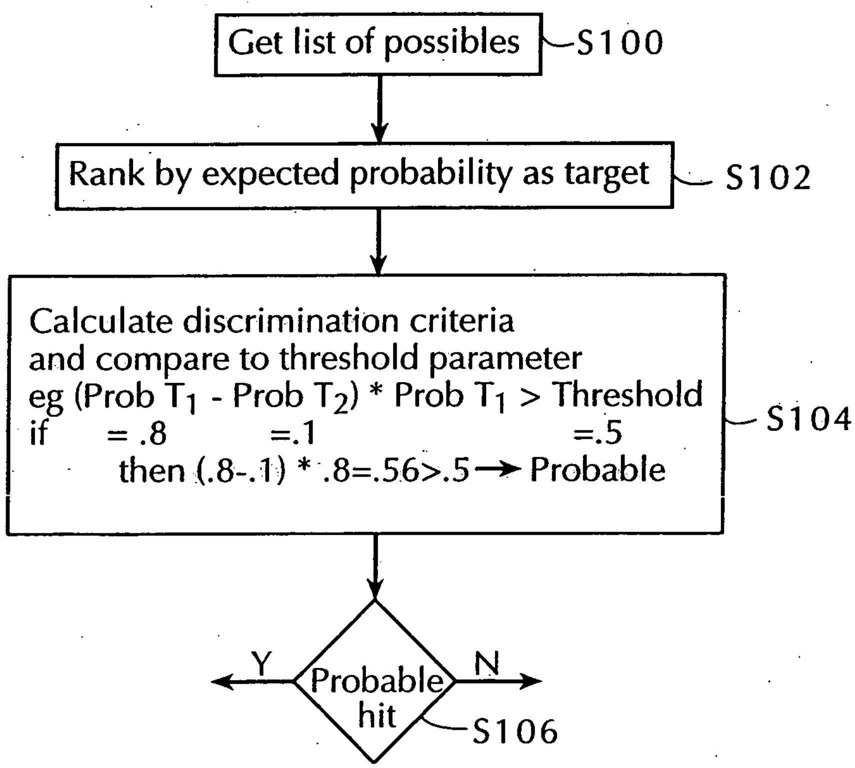

S40
N
Selected?
Y
S44
increment raw score by factor$_S$
S42
N
Correct?
Y
S46
increment raw score by factor$_{Y'}$
No, feedback
decrement raw score by factor$_{0'}$
(can be zero)
S50
S48
Decrement
raw score
by factor$_{N'}$
increase experience level E Factor$_{ps'}$, E Factor$_{pc}$
S52

FIG. 5



Get list of possibles
S100
Rank by expected probability as target
S102
Calculate discrimination criteria
and compare to threshold parameter
eg (Prob T_1 - Prob T_2) * Prob T_1 > Threshold
if = .8 =.1 =.5
then (.8-.1) * .8=.56>.5 → Probable
S104
Probable hit
Y
N
S106

TITLE

METHOD AND APPARATUS FOR UTILIZING USER FEEDBACK TO IMPROVE SIGNIFIER MAPPING

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention is directed to a computer-implemented product for locating and connecting to a particular desired object or target resource from among plural resources resident at distributed locations on a network.

Description of the Related Art

The worldwide network of computers known as the Internet evolved from military and educational networks developed in the late 1960's. Public interest in the Internet has increased of late due to the development of the World Wide Web (hereinafter, the Web), a subset of the Internet that includes all connected servers offering access to hypertext transfer protocol (HTTP) space. To navigate the Web, browsers have been

developed that give a user the ability to download files from Web pages, data files on server electronic systems, written in HyperText Mark-Up Language (HTML). Web pages may be located on the Web by means of their electronic addresses, known as Uniform Resource Locators (URLs).

A URL uniquely identifies the location of a resource (web page) within the Web. Each URL consists of a string of characters defining the type of protocol needed to access the resource (e.g., HTTP), a network domain identifier, identification of the particular computer on which the resource is located, and directory path information within the computer's file structure. The domain name is assigned by Network Solutions Registration Services after completion of a registration process.

While the amount of information available on the Web is enormous, and therefore potentially of great value, the sheer size of the Web makes the search for information, and particular web sites or pages, a daunting task. Search engines have been developed to assist persons using the Web in searching for web pages that may contain useful information.

Search engines fall into two major categories. In search engines falling into the first category, a service provider compiles a directory of Web sites that the provider's editors believe would be of interest to users of the service. The Yahoo site is the best known example of such a provider. Products in this category

are not, strictly speaking, search engines, but directories, and will be referred to hereinafter as "editor-controlled directories". In an editor-controlled directory, the developer of the directory (the "editor") determines, based upon what it believes users want, what search terms map to what web pages.

The other major category, exemplified by Altavista, Lycos, and Hotbot, uses search programs, called "web crawlers", "web spiders", or "robots", to actively search the Web for pages to be indexed, which are then retrieved and scanned to build indexes. Most commonly, this is done by processing the full text of the page and extracting words, phrases, and related descriptors (word adjacencies, frequencies, etc.). This is often supplemented by examining descriptive information about the Web document contained in a tag or tags in the header of a page. Such tags are known as "metatags" and the descriptive information contained therein as "metadata". These products will be referred to hereinafter as "author-controlled search engines," since the authors of the Web documents themselves control, to some extent, whether or not a search will find their document, based upon the metadata that the author includes in the document.

Each type of product has its disadvantages. Author-controlled search engines tend to produce search results of enormous size. However, they have not been reliable in reducing the large body of information to a manageable set of relevant results. Further, web site authors often attempt to skew their site's position in

the search results of author-controlled search engines by loading their web site metatags with multiple occurrences of certain words commonly used in searches.

Editor-controlled directories are more selective in this regard. However, because conventional editor-controlled directories do not actively search the web for matches to particular search terms, they may miss highly relevant web sites that were not deemed by the editors to be worthy of inclusion in the directory. Also, it is possible for the editor to "play favorites" among the multitude of Web documents by mapping certain Web documents to more search terms than others.

Recently, search engines such as DirectHit (www.directhit.com) have introduced feedback and learning techniques to increase the relevancy of search results. DirectHit purports to use feedback to iteratively modify search result rankings based on which search result links are actually accessed by users. Another factor purportedly used in the DirectHit service in weighting the results is the amount of time the user spends at the linked site. The theory behind such techniques is that, in general, the more people that link on a search result, and the longer the amount of time they spend there, the greater the likelihood that users have found this particular site relevant to the entered search terms. Accordingly, such popular sites are weighted and appear higher in subsequent result lists for the same search terms. The Lycos search engine (www.lycos.com) also uses feedback, but only at the time of crawling, not in

ranking of results. In the Lycos search engine, as described in U.S. Patent No. 5,748,954, priority of crawling is set based upon how many times a listed web site is linked to from other web sites. This idea of using information on links to a page was later exploited by the Clever system developed in research by IBM, and the Google system (www.google.com), which do use such information to rank possible hits for a search query.

Even leaving aside the drawbacks discussed above, search engines of both categories are most useful when a user desires a list of relevant web sites for particular search terms. Often, users wish to locate a particular web site but do not know the exact URL of the desired web site. Conventional search engines are not the most efficient tools for doing this.

Moreover, naming and locating particular sites on the Web is currently subject to serious problems. For example, appropriate names, including existing company names or trademarks, may not be available, because someone registered them first. Names may be awkward and not obvious, because of length, form/coding difficulties or variant forms, and names may not justify a separate domain name registration for reasons of cost and convenience, such as movie titles or individual products.

This problem results from a mismatch between the present network addressing scheme based on Uniform Resource Locators (URLs), which meet the technical

needs of the Internet software, and the needs of human users and site sponsors for simple, user-friendly mnemonic and branded names. This problem is largely hidden in cases where a user finds a site by clicking a pre-coded link (such as after using a search engine), or by using a saved bookmark. However, the problem does seriously affect users wishing to find a site directly, or to tell another person how to find it. To do this, the person must know and type the URL into his Internet browser, typically of the form sitename.com or www.sitename.com. Site sponsors are also seriously hampered by this difficulty in publicizing their sites.

Further, the current method of naming and locating Web sites has serious, widely known problems. Web site locator "domain" names are often not simple or easily remembered or guessed, and often do not correspond to company, trademark, brand or other well-known names.

As a result of the foregoing, site URLs (or domain names) are not intuitively obvious in most cases, and incorrect access attempts waste time and produce cryptic error messages that provide no clue as to what the correct URL might be. A significant percentage of searches are for specific, well-known sites. These could be found much more quickly by a special-purpose locator engine. The current mode of interacting with search engines is also cumbersome-for this purpose, a much simplified mode of direct entry is practical.

One attempt to provide the ability to map a signifier, or alias, to a specific URL utilizes registration of

key words, or aliases, which when entered at a specified search engine, will associate the entered key word with the URL of the registered site. One such commercial implementation of this technique is known as NetWord (www.netword.com). However, the NetWord aliases are assigned on a registration basis, that is, owners of web sites pay NetWord a registration fee to be mapped to by a particular key word. As a result, the URL returned by NetWord may have little or no relation to what a user actually would be looking for. Another key word system, RealNames (www.realnames.com), similarly allows web site owners to register, for a fee, one or more "RealNames" that can be typed into browser incorporating RealNames' software, in lieu of a URL. Since RealNames also is registration based, there is no guarantee that the URL to which is user is directed will be the one he intended.

Further, in existing preference learning and rating mechanisms, such as collaborative filtering (CF) and relevance feedback (RF), the objective is to evaluate and rank the appeal of the best *n* out of *m* sites or pages or documents, where none of the *n* options are necessarily known to the user in advance, and no specific one is presumed to be intended. It is a matter of interest in any suitable hit, not intent for a specific target. Results may be evaluated in terms of precision (whether "poor" matches are included) and recall (whether "good" matches are not included).

A search for "IBM" may be for the IBM Web site, but it could just as likely be for articles about IBM as a

company, or articles with information on IBM-compatible PCs, etc. Typical searches are for information *about* the search term, and can be satisfied by any number of "*relevant*" items, any or all of which may be previously unknown to the searcher. In this sense there is no *specific* target object (page, document, record, etc.), only some open ended set of objects which may be useful with regard to the search term. The discovery search term does not signify a single intended object, but specifies a *term* (which is an attribute associated with one or more objects) presumed to lead to any number of relevant items. Expert searchers may use searches that specify the subject indirectly, to avoid spurious hits that happen to contain a more direct term. For example, searching for information about the book Gone With The Wind may be better done by searching for Margaret Mitchell, because the title will return too many irrelevant hits that are not about the book itself (but may be desired for some other task).

In other words, the general case of discovery searching that typical search engines are tuned to serve is one where a search is desired to return some number, n, of objects, all of which are relevant. A key performance metric, *recall*, is the completeness of the set of results returned. The case of a signifier for an object, is the special case of n=1. Only one specific item is sought. Items that are not intended are not desired--their relevance is zero, no matter how good or interesting they may be in another context. The top DirectHit for "Clinton" was a Monica Lewinsky page. That is probably not because people searching for

Clinton actually intended to get that page, but because of serendipity and temptation—which is a distraction, if what we want is to find the White House Web site.

In addition,

-CF obtains feedback from a group of users in order to serve each given user on an overall, non-contingent basis--without regard to the either the intent of the user at a specific time, or to being requested in a specific context.

-RF is used by a single user to provide feedback on their intent at a given time, but still with no presumed intent of a single target.

More broadly, searching techniques are generally not optimized based on using a descriptor which is also an *identifier*--they provide more generally for the descriptor to specify the nature of the content of the target, not its name. There are options in advanced search techniques which allow specification that the descriptor is actually an identifier, such as for searching by title. Such options may be used to constrain the search when a specific target happens to be intended, but no special provision is made to apply feedback to exploit that particular relationship or its *singularity*.

Moreover, none of the currently available key word systems utilize heuristic techniques actually to determine the site intended by the user. Instead, the current systems teach away from such an approach by their use of registration, rather than user intention,

to assign key words to map to web pages. Thus, the current techniques are not directed to solving the problem of finding the one, correct site for a particular signifier.

Thus, the need exists for a system that would enable a user to find a desired Web document by simply entering an intuitive key word or alias and that would perform a one to one mapping of the alias with the URL actually desired by the user, and which would use heuristic techniques to assist in providing the correct mapping, and improving system accuracy over time.

SUMMARY OF THE INVENTION

In consideration of the above deficiencies of the prior art, it is an object of the present invention to provide a method of signifier mapping that allows a user to locate to a particular network resource, in the preferred embodiment a web page, by simply entering a signifier or alias.

Thus, the present invention is generally directed to a technique for intelligent searching or matching where a *signifier* is given and is to be related to a name or address of an *intended* target object.

Signifier, in the context of the present invention means:

-an identifier, referent, or synonym for the name or address of a specific resource (a target object) presumed to exist in some domain; but

-not necessarily a "name" or "address"--a canonical identifier that has been assigned by some authority or pre-set by some convention (names are a subset of signifiers--those which are canonical or pre-established);

-not necessarily a description of content or subject matter (concepts or words);

-an identifier that has cognitive significance to the user, and presumed communication value in identifying the intended target object to another person or intelligent agent.

In addition, this cognitive/communication value is based on a perceived relationship (meant to have minimal ambiguity) to an identifier, which might be an assigned name or a name based on common usage, but which need not be exact, as long at it serves to signify the intended target.

More generally, descriptors may possibly be considered to be signifiers, if they are intended to be unique or minimally ambiguous (e.g. "the company that commercialized Mosaic" or "the company that sells the ThinkPad").

It is a further object of the present invention to provide a system in which heuristic techniques are used together with user feedback to improve the accuracy of signifier mapping.

None of the many solutions to the signifier mapping problem (Netword, Centraal, Goto, etc.) have identified

learning as a valuable technique. This may be because what naturally come to mind are techniques based on pre-defined mappings that make the use of "de jure" explicit registration. That teaches away from the idea of trying to learn the mappings heuristically from colloquial usage. (The same applies to attempts at creating systems for "user friendly names" in other directory systems.) Since the mappings are understood as being defined or registered, why would one try to learn about them? But actually, the mappings are just like natural language—they are dynamic, evolving, and ambiguous, and can only be resolved in terms of *learned usage within a context* — which is best addressed by learning, as in the present invention, not registration or other static mappings as appear in the prior art.

The use of heuristic, adaptive feedback-based techniques operates in significantly different ways when focused on signifier mapping, and this can be exploited by isolating such tasks. ~~The~~ A key difference between the present invention and most common searching tasks is that in the prior searching techniques, there is no intention of a specific target object that is known to exist.

The present invention has several advantageous features, various combinations of which are possible:

1) a special purpose mapping engine for locating popular sites by guessed names;
2) automatic display of the target site (if located with reasonable confidence);

3) an optional simplified mode of direct entry of a guessed site name; and

4) use of user expectations, such as popularity of guesses intended for a given site, as a primary criterion for translating names to sites, with provision for protection of registered trademarks or other mandates.

In accordance with one aspect of the present invention, a finder or locator server is established. The server is configured to work with a user interface that allows users to enter an guessed name or alias, as easily as if the user knew the correct URL for the intended target resource. In response to entry of the alias, the finder server accesses a database that includes, in a preferred embodiment, conventional Web-crawler-derived index information, domain name registration information, as well as user feedback from previous users of the server, and looks up the correct URL, i.e., the one URL that corresponds to the alias and causes the user's browser to go automatically to that URL, without the user having to view and click on a search results page, if the correct URL can be determined with a predetermined degree of confidence.

In one preferred embodiment, the server is structured to accept the alias as a search argument and do a lookup of the correct URL and the return of same to the browser, without the intermediate step of the user having to wait for and then click on a search results web page. The automatic transfer is preferably effected using standard HTML facilities, such as a

redirect page or framing. Redirect is effected by placing pre-set redirection pages at the guess URL on the server. Alternately, the redirect page can be generated dynamically by program logic on the server that composes the page when requested.

The present invention advantageously uses feedback and heuristic techniques to improve the accuracy of the determination of the correct URL. If a suggested match is found by the look-up technique and the accuracy of the mapping is confirmed by user feedback, then, after directing the user to the URL, the result is stored in the server to improve the accuracy of subsequent queries. The server database includes a list of expected terms and expected variants that can initially be catalogued to provide for exact matches. This list is updated by the learning processes discussed in more detail below.

If, on the other hand, a probable one intended match cannot be determined, the finder server preferably uses intelligent techniques to find a selection of links to possible matches ranked in order of likelihood, or could return a no-match page with advice, or a conventional search interface or further directories.

According to a preferred embodiment of the invention, each of the selection of links are configured not to go directly to the target URL. Rather, the links are directed back to a redirect server established by the finder server, with coding that specifies the true

target, and feedback information. The finder server can in this way keep track of user selections.

In accordance with an advantageous aspect of the invention, such feedback information is used to improve the results of the search by promoting web sites almost universally selected to exact match status, and by improving the ranking of possible lists in accordance with which links are most often selected. Preferably, a confidence parameter can be generated from such tracking to control whether to redirect to a URL or to present a possible list to users.

In furtherance of the above and other objects, there is provided, a designated server, accessible on the Internet, the designated server being configured to respond to relocation requests that specify an identifier, corresponding to a target resource, that may not be directly resolvable by standard Internet Protocol name resolution services to the URL of the target resource. In a direct entry embodiment of the present invention, requests are passed to the relocation server by sending a relocation URL that designates the relocation server as the destination node and appends the identifying information for the identifier as part of a URL string. The relocation server extracts the identifying information and translates it into a valid URL for the target resource. The relocation server is configured, in the event that a unique URL can be determined with respect to the target resource, to cause the target resource to be

presented to the user without further action on the part of the user.

Preferably, the user requests are entered at a web browser, the relocation or search server determines the valid URL for the target resource by performing a look-up in a database, and the response from the relocation server is in the form of a redirect page that causes the user's web browser to obtain the target resource.

In accordance with one aspect of the present invention, there is provided a method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The method is for use on a finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method by the user and plural previous users; and (b) a learning system structured to access and learn from information contained in the database. The method comprises: receiving a resource identity signifier from the user; and accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, the method further comprises directing a computer of the user so as to enable that computer to connect the user to the

address of the resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided an apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered in previous operations of the apparatus by a user and plural previous users; and (b) a learning system operable to access and learn from information contained in the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by: receiving a resource identity signifier from the user; and accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, a computer of the user is directed so as to cause that computer to connect the user to the address of the resource, if any, determined to be the intended target resource.

In accordance with yet another aspect of the present invention, there is provided a system for finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource

identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The system comprises: finder server means having: (a) database means for storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system by the user and plural previous users; and (b) learning system means for accessing and learning from information contained on the database; receiving means for receiving a resource identity signifier from the user; and accessing means for accessing the database means to determine which, if any, of the indexed resources is likely to be the desired target resource. Preferably, the system further comprises directing means for directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined in the access means to be the target resource.

In accordance with still another aspect of the present invention, there is provided a computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server by a user and plural previous users; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource

intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers. The method comprises: receiving a resource identity signifier from the user; and accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource. Preferably, the method further comprises the step of: directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided a system for finding resources on a network of interconnected computers on which a plurality of resources reside. The system comprises: a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and a finder server having: (a) a database including: (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous operations of the system by the user and plural previous users; and (b) a learning system operable to access and learn from information contained in the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of

resources located on the network, by: receiving a resource identity signifier from the user; accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

In accordance with another aspect of the present invention, there is provided a method of identifying, in response to entry by a user of an object identity signifier, a single intended object to be acted upon, the single intended object being intended by the user to uniquely correspond to the object identity signifier, among a plurality of possible objects. The method is for use on a computer having: (a) a database including (i) an index of possible objects; and (ii) information regarding user feedback gathered in previous executions of the method by the user and plural previous users; and (b) a learning system structured to access and learn from information contained in the database. The method comprising: receiving an object identity signifier from the user; and accessing the database to determine, based upon the information in the database, which, if any, of the indexed objects is likely to be the object intended to be acted upon.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A is an architectural block diagram of a server computer system internetworked through the Internet in accordance with a preferred embodiment of the present invention;

Figure 1B is a flow diagram illustrating a method of obtaining feedback from multiple users to be applied in searching or signifier mapping;

Figure 2 is flow diagram showing a method of signifier mapping using feedback and heuristics to continually improve the performance of the mapping;

Figure 3 shows an example of a database entry for the finder server of the present invention;

Figure 4A is a flow diagram illustrating a technique of feedback weighting for probable results in signifier mapping;

Figure 4B is a flow diagram illustrating a technique of feedback weighting for possible results in signifier mapping; and

Figure 5 is a flow diagram illustrating how feedback is used in a preferred embodiment to discriminate a probable target resource in accordance with the present invention.

DETAILED DESCRIPTION OF THE DISCLOSED EMBODIMENTS

"Population cybernetics" and the Internet

As a general matter, the present invention relates to a technique that collects experience (a knowledge base) from a mass population that is open ended or universal, either over all domains, or over some definable subject or interest domain or strata. This represents a significant improvement over prior art techniques, which are generally limited in the scope of the population and extent of experience from which they draw their knowledge base.

The technique of the present invention, in a preferred embodiment, uses the Internet to do this in a way that is powerful, economical, and far-reaching. The technique, in the preferred embodiment, uses the Internet to enable collection and maintenance of a far more complete knowledge base than has been used with any prior technique except Collaborative Filtering (CF).

In the present invention feedback learning is advantageously utilized, so that the information is not just collected, but refined based on feedback on the accuracy of prior inferences.

In its broad sense the present invention constitutes a kind of "population cybernetics," in that the learning does not just collect a linear knowledge base, but uses a feedback loop control process to amplify and converge

it based on the results of prior inferences, and that it works over an entire population that is open, infinite, and inclusive. This is in contrast to prior learning techniques, which draw on necessarily finite, closed populations.

The use of population group information to achieve signifier mapping differs from the prior art technique of collaborative filtering in at least the following manner:

Whereas both CF and the technique of the present invention draw on knowledge of a population group to make inferences, CF obtains *ratings* of many things by many people to suggest other things (that may also be highly rated by the user, based on correlation with the group), and CF *does not involve* a *specific input request*, but rather seeks a *new, previously unknown item* in a category. On the other hand, the present invention obtains *translations* of many signifiers by many people to suggest the intended translation of a signifier and *involves a specific input request* to be translated to identify a *known intended target*

Although the technique of signifier mapping will occasionally be referred to loosely as searching, it is more accurately translation, because the target is intended and known, just not well specified. This differs from typical Web or document searching, which typically seeks unknown, new items.

The technique of the present invention also differs from natural language (NL) translation or understanding, in that the input ~~is *atomic* in that it~~ has no context as part of a body of discourse (~~a~~ i.e., text). NL understanding techniques on the other hand translate words as components of concepts embedded in texts having a *context* of related ideas. Thus the cues of context in a discourse are absent, and the translation must be done without any such cues, although knowledge of the user may provide a useful context of behavior, demographics, psychographics that has some value in inferring intent, and knowledge of the user's prior requests may provide additional useful context information. The task is to infer or predict intention, rather than to understand meaning, because there is no basis to infer meaning in any conceptual sense. The input is disjointed from any surrounding context, and if not seen before (from the user or others), there is little useful information on either its meaning or intention. The present invention seeks to infer intention based on limited data, primarily the input request, and draws on group data (of request translations) as its strength.

The task of the present invention has similarities with cryptanalysis, in that both the present invention and cryptanalysis use data about communications behavior from groups of communicators to make inferences. However the task differs in that

- Cryptanalysis deals with intentional hiding of meaning or intention, where the technique of the present invention is applied to cases where the

hiding (of the intention of a signifier) is not at all intended; and

- Cryptanalysis seeks to infer meaning (ideas) drawing on context in a discourse, like NL understanding, not usually to infer the intention of a signifier (of objects or actions) which is not in a context.

This point of intention versus meaning is subtle, but has to do with communication of commands or requests as opposed to concepts.

- One view of this is the idea of "requests," as opposed to declarations or assertions, in the use of language.
- This task of recognizing commands (vs. meanings) has parallels in the task of robot control, such as that based on spoken commands. The similarity is in training understanding of the speech of many users to be speaker independent, and to infer meanings of a current speaker from that of others. The difference is that the tasks addressed in the present invention deal with a very wide, effectively infinite universe of commands (intended objects), while robot control techniques have generally been limited to very small sets of commands (partly because of the inability to apply mass experience).

Thus the technique of the present invention could be viewed as addressing a special class of robot control (in which experience data and feedback is accessible), and may ultimately be extensible to other robot control

applications as such data becomes accessible over the network.

The social dimension is critical for inferences relating to shared objects or resources. Names draw on social conventions and shared usage. This social usage information is essential to effective mapping of signifiers to resources. De-jure naming systems can underlie a naming system, as for current Internet domain names, but de-facto usage is the essential observable source of information for fullest use. De-jure systems suffer from entropy, corruption and substitution, while de-facto usage is pragmatic and convergent to changing usage patterns.

This applies to a variety of name-able resources:

- Web domain names;
- Web sub-site names (such as to find sub-areas);
- People or business names;
- Department, agent, or service identifiers (such as to to find contact points);
- Policy capability specifications (such as to find permissions, such as someone who can provide access to a given resource for a given purpose, such as confirming employment status or update-access to a report);
- Information sets or collections (to find reference tools that are known to exist, such as an IBM dictionary of acronyms, or an index of papers in ACM publications);
- Other robot control tasks, as social experience and feedback becomes accessible.

Social usage information can be combined with other sources of information in a heuristic fashion. For example, there could be a hierarchy that might be used in order, as available:

1. Personal defined usage information, such a defined personal nicknames;
2. Public de-jure defined mappings or directories;
3. Personal usage information (a person's own undefined nicknames, learned from that person's own usage/feedback);
4. Social de-facto usage information;

This is just one possible sequence, but shows how the usage data can take searching beyond what has been defined.

As discussed above, a preferred embodiment of the present invention relates to a method and apparatus for locating a desired target resource located and accessible on a network, in response to user entry of a guessed name or alias. In illustrating the preferred embodiment, the apparatus is shown as a server computer, or computers, located as a node on the Internet. However, the present invention is in no way limited to use on the Internet and will be useful on any network having addressable resources. Even more broadly, the present invention is useful for any similar task of identifying an intended target for an action in which automatic facilitation of that action is desired, where feedback from a large population can be obtained to learn whether a given response was in

fact the one that was desired. Control of robots, as discussed herein, is one example of such broader application.

The finder server of the preferred embodiment of the present invention allows users to enter a guessed identifier or alias, as easily as if they knew the correct URL. Specifically, the finder server of the present invention accepts a guessed name, or alias, from a user, uses a look-up technique, enhanced by heuristics preferably taking into account previous users' actions, to determine a correct URL for the intended target resource, and causes the user's browser to go to that URL automatically. Preferably this is done without the added step of first viewing and clicking on a search-results page, where an initial search finds the intended target resource with a predetermined degree of certainty. Such a resource will be referred to hereinafter as a "probable". In accordance with a preferred embodiment of the present invention, this functionality can be implemented by:

- Publicizing the locator server under an appropriate URL name, for example, guessfinder.com.
- Setting up the server to, in response to entry of a guessed name or alias, do a lookup to the correct URL and return a response that causes the user's browser to go automatically to the specified URL. Such an automatic transfer can be effected using a standard HTML facilities, such as a redirect page, or framing.
- If the guess does not provide an exact match in the lookup phase, using feedback and heuristic techniques to create and present to the user a selection of

links to possible matches. Alternately, the user may be presented with a nomatch page with advice, or directed to a conventional search interface, or further directories.

It is contemplated that the use of aliases for attempting to locate a web site associated with company name or brand name would be found useful. For example, the aliases "s&p", "s-p", "sandp", "snp", "standardandpoors", "standardnpoors", "standardpoors" should preferably all map to www.standardpoors.com. In addition to companies and brands, other important name domains would include publications, music groups, sports teams, and TV shows.

The present invention advantageously provides for learning and feedback on the basis of user preferences to automatically and dynamically build a directory of names and sites that maps to the actual expectations and intentions of a large population of users, and adapts to changes over time, including the appearance of new sites, thus optimizing utility to them.

The finder server of the present invention effectively provides a secondary name space, administered by the organization operating the finder system, through the automated heuristic methods described here, that maps to, but is not dependent on, the URL name space. The finder site computer has access to a data base containing entries for any number of popular sites, with any number of likely guesses and variations for each site.

As a result of the service provided when the present invention is implemented, site sponsors could skip the cumbersome and costly process of obtaining specific mnemonic URLs or alternate URLs in many cases (especially with regard to domain names). Even with a number of conventional URLS, this service could be a supplement, for additional variations. The problem of pre-empted URL domain names would also be avoided, except where there is legitimate and significant pre-existing usage.

A key to utility is to be able to directly connect in response to most guesses, and ambiguities could be a limiting factor. To avoid that it is desirable to exploit Pareto's Law/the 80-20 rule and do a direct connect even when there is an uncertain but likely target. For that to be useful, it must be easy for users to deal with false positives.

Correction after arrival at a wrong site can be made relatively painless by allowing a subsequent request to indicate an error in a way that ties to the prior request and adds information. For example a request, guessfinder.com/lionking, that located the movie but was meant to find the play could be corrected by entering guessfinder.com/lionking/play. A more efficient coding might explicitly indicate an error, such as guessfinder.com/!/lionking/play. Even with the error, this would be quicker and easier than conventional methods. Note that this example was illustrated with the direct URL coding techniques described below. Similar post-arrival corrections can

be made with other user interface techniques, such as a frame header that includes appropriate user interface controls to report feedback, much as conventional search engines allow for "refinement" of prior searches, also described below.

Correction in-flight can be achieved by using the existing visibility of the redirect page, or enhancing it. When a redirect page is received by a user's browser, it appears for a short time (as specified with an HTML refresh parameter) while the target page is being obtained. In addition to affording a way to optionally present revenue-generating (interstitial) advertising content, that page preferably lists the redirection target, as well as alternatives, allowing the user to see the resolution in time to interrupt it. This is most useful with a browser that permits a redirect to be stopped in mid-stream by clicking the stop button, leaving the redirect page on display, and allowing a correct selection among alternative links to be made. Alternately, a multi-frame (multi-pane) display could be used to allow a control frame to remain visible while the target page is loading in a results frame, as described below.

Note some of the typical parameters and control points that would be relevant:

"New" sites.

Applies when the user wants a site but is provided neither a direct hit, nor a correct possible. Users would find the site via alternate means (offered through the service or not). The user then submits an add-site request, via the Web or e-mail. If the number of add-site requests over a set interval exceeds a set (low) threshold, the site is added as a possible, or a direct hit if there are no competing alternatives. Such adds would be provisional, and could be dropped if requests are not sustained.

Possibles

Low confidence possibles would be listed low on the list, and selections would be tracked. If selections are strong, they move up the list. If selections are very weak, they would drop off after some interval. The threshold to add back sites that were dropped might be higher for a time, to limit oscillation and false adds. If possibles are well ahead of alternatives by some threshold over some interval, they would be promoted to direct hits.

Direct hits

Feedback on false positives would be collected. This could be via links in frames, redirect pages, interstitials, or other means, as suggested previously. If false positives exceed a threshold, the site would revert to a possible and the common alternatives would be listed as well.

Parameter issues: thresholds, intervals, smoothing, damping, overrides.

Basic parameters include the various thresholds and time intervals for measurement. Smoothing techniques (such as exponential smoothing) would be applied to adjust for random variations and spikes, to improve forecasting. Damping mechanisms could be used to limit undue oscillation from state to state. Overrides would provide for mandated or priority matches, such as for registered trademarks, on either a weighted or absolute basis, as appropriate.

Figure 1A illustrates a first embodiment of the present invention, as implemented on the Internet. The finder server 10 includes a computer or computers that perform processing, communication, and data storage to implement the finder service. Finder server 10 includes a finder processing/learning module 101. Module 101 performs various processing functions, and includes a communication interface to transmit and receive to and from the Internet 12, as well as with database 102, and is programmed to be operable to learn from experiential feedback data by executing heuristic algorithms. Database 102 stores, in a preferred embodiment, indexes of URL data that would allow the module 101 to locate, with a high degree of confidence, a URL on the Web that is an exact match for a target resource in response to a user's entry of an alias or guessed name. Preferably, the indexes store, in addition to available URL information, such as domain name directories, information relating to the

experience of the server in previous executions of the finder service. As the server gains experience and user feedback, heuristic techniques are applied by module 101 to enable the returned URLs to conform more and more accurately to user expectations.

Users 11_0-11_N can access the Internet 12 by means of client computers (not shown) either directly or through an Internet service provider (ISP). As has been discussed previously, to make use of the present invention the user enters a guessed name, or alias, into his computer's browser and submits a query containing the alias to the finder server. The World Wide Web 14 includes computers supporting HTTP protocol connected to the Internet, each computer having associated therewith one or more URLs, each of which forming the address of a target resource. Other Internet information sources, including FTP, Gopher and other static information sources are not shown in the figure.

The finder server includes operating system servers for external communications with the Internet and with resources accessible over the Internet. Although the present invention is particularly useful in mapping to Internet resources, as was discussed above, the method and apparatus of the present invention can be utilized with any network having distributed resources.

Entry of the alias by a user may be accomplished in a number of ways. In one embodiment, a usage convention can be publicized for passing the alias to the server

within a URL string, such as guessfinder.com/get?ibm, for example, for trying to find the web page corresponding to the alias "ibm". In this case, the server is programmed to treat the string "ibm" as a search argument and perform the appropriate processing to map the alias to the intended target resource. A similar effect can be obtained by the somewhat simpler form guessfinder.com/ibm, if the server is programmed appropriately. Alternately, the user can visit the web site of the finder server and be presented with a search form, as is done in conventional search engines. A third option is to provide a browser plug in that allows direct entry of the key word in the browser's URL window or any alternative local user interface control that will then pass the entry on as a suitably formatted HTTP request.

It also would be preferable for an enhanced user interface to be phased in as the service gains popularity. This preferably would be accomplished by a browser plug-in, or modifications to the browser itself, to allow the alias to be typed into the URL entry box without need for the service domain name prefix (such as, guessfinder.com/...). Instead, such an entry would be recognized as a alias, not a URL, and the prefix would be appended automatically, just as http://... is appended if not entered with a URL in current browsers.

Figure 1B is a flow diagram illustrating a technique for obtaining and learning from feedback responses gathered from a large group of people, in the example,

users 1, 2, . . . n. Such a technique can be used in a variety of applications, and in particular in traditional search engines, or in mapping to identify particular web sites, as in alias or signifier mapping.

In Figure 1B, users 1, 2, . . . n represent a large community of users. In the flow diagram, the flow of query items from the users is indicated by a Q, the flow of responses back to the users is indicated by an R, and the flow of feedback results provided by the users' actions, or responses to inquiries, is indicated by an F. As can be seen from the figure, Query (a, 1) is transmitted from user 1 to the service 2, which can either be a searching or a mapping service. The service has learning processor 4, which interfaces with a database 6. The database 6 contains, among other things, indexes and feedback information gathered from previous queries. In response to the query, the user 1 is provided with a response R(a, 1). User 1 then is provided with the opportunity to transmit user Feedback (a, 1) to the Service 2. Learning processor 4 stores the feedback information in the database 6, and is programmed with one or more heuristic algorithms enabling it to learn from the feedback information to improve the returned search or mapping results. The feedback provided will improve the results offered, for example by positively weighting results preferred by users, so that, over time, more accurate results can be obtained.

Figure 2 is a diagram illustrating the logical flow used in applying the general technique of learning from

user feedback shown in Figure 1 to signifier mapping, in accordance with a preferred embodiment of the present invention. A user enters a Query consisting of a signifier, represented by Q^s. The server, in response to receipt of the query, parses the query, at step S02, and in step S04 performs a database lookup in an attempt to determine, if possible, the exact target resource intended by the user. Database 6 includes index data as well as feedback data obtained from users in previous iterations of the signifier mapping program, is accessed. The stored data structure is described in more detail below.

In step S06, the program discriminates a probable intended target making use of the index information such as domain registration indexes, and other resources, as well as the feedback information stored in the database. In step S08, if a likely hit, or exact match has been identified, that is, a web page has been located with a high confidence parameter, the flow continues to step S10. At step S10, a direction is prepared to the likely hit URL. A list of alternatives optionally may be provided for presentation to the user at the same time, in case the likely hit turns out not to be the target identifier. At step S12, the server sends information R^s to the user, more particularly to the user's browser, to effect a link to the likely hit. Optionally, the alternate list is also provided at the same time.

In step S14, the viewed page is monitored by the server and the user, by his actions, provides feedback. Most

readily determined with no assistance from the user is the fact of the user having chosen the link. This may be determined, for example by a redirect, in which an intermediate server is transparently interposed between the browser and the target page, and thus able to identify the user and the URL target based on coding built into the URL that the user clicks. Also desirable is the amount of time the user spends at the site, which will be an indicator of whether the site is the intended target. This may be ascertained, for example, if clickstream data can be obtained, such as through the use of a monitor program that works as a browser add-in or Web accessory, such as the techniques offered by Alexa. Other feedback can be provided by asking the user. This can, for example, be done conveniently by using a small header frame served by the relocation service that appears above the actual target page, and that includes controls for the user to indicate whether or not the results were correct. The URL of the viewed page is recorded, together with any other feedback, for use in improving the accuracy of subsequent iterations of signifier mapping. At step S26, the feedback data is supplied to a feedback weighting algorithm, described in detail below, which generates appropriate weighting factors to be stored in the database for use in subsequent mappings.

If it is determined at step S08 that the result is not a likely hit, the flow proceeds to step S18, where a list of the top m hits (m being a predetermined cutoff number), preferably drawing on the list of possible hits from a conventional search engine, or by employing

the same techniques as a conventional search engine, is prepared. Unlike conventional search engines, the ranking of these hits is based primarily on experience feedback data as described below. In addition, where such feedback is limited or absent, it would be supplemented by variants of more conventional search engine weighting rules that are expressly tuned to the task of finding a single intended result (i.e., high relevance by low recall) rather than many results (high relevance plus high recall). The list is presented, at step S20, to the user as R^s. The user, by the selections made from the provided list, and from other feedback, such as how long the user spends at each link, supplies feedback to the system. This information F^s is monitored, at step S22 and recorded, at step S24. The recorded information is supplied to the feedback weighting algorithm, at step S26, the output of which is stored in the database for use in subsequent iterations of the signifier mapping.

Figure 2 illustrates the simple case in which a user is directed to a target URL if the target has been determined to be a probable hit, and is presented with a list to choose from if the target cannot be identified with sufficient certainty. However, it is well within the intended scope of the invention for alternate methods to be employed. For example, the user interface (UI) could be extended, either by framing, or a browser plug-in or extension, to provide multi-pane/multi-window results that allow a pane for each type of response, e.g., the target response and a list of possibles, regardless of the level of

confidence in the result. In such a case, the format for presentation of results would be the same whether a probable has been located or not, but the learning from feedback and ranking would still seek to determines "correctness" based on the varying feedback cases.

Figure 3 illustrates a preferred method of organizing index data to allow for storing and updating of the most probable hits for a given query. As can be seen from the illustration, for each query, whether single element queries or compound queries, there is stored a list of associated possible targets. Linked to each of these query/target pairs is a raw score, an experience level, and a probability factor. As feedback enters the system, the index data is updated to reflect the user feedback. The updating process will be described below. While the index shows preferred weighting criteria, these are only a sample of the kind of criteria that can be correlated to the query/target pairs. In a simple embodiment, the raw score would be based only on selections of hits, and explicit feedback on correctness as described below. Other embodiments could add feedback data on time spent at a target. Additional variations would include weighting based on the recency of the feedback, and on the inclusion of non-feedback data, such as the various syntactic and semantic criteria used for relevance weighting by conventional search engines.

The process of maintaining the guess-target database is adaptable to a high degree of automation, and this can

be highly responsive to new sites. An outline of such a method is:

All guesses are logged and analyzed.

Ambiguous hits are tracked as described earlier.

Complete non-matches are sorted by frequency to identify common new requests (in real time). Changes in ambiguous match patterns could also flag appearance of new sites.

Common new requests preferably are fed to an automated search tool that would use existing search engines, hot site lists, and name registration servers, etc. to identify possible targets.

Automated intelligent analysis of those results can seek to qualify probable targets.

High confidence (or possible) targets preferably are added, and then tracked based on the feedback mechanism described earlier, in order to self-correct. A confidence parameter preferably is used to control whether to redirect or to present a possibles list to users.

Human review and correction also preferably is used to supplement this.

Figure 4A illustrates a preferred technique for weighting the results using feedback data for hits that have been determined to be probable hits. In step S30, if the user feedback from the probable result indicates that the probable was in fact the target URL the user was searching for, the flow proceeds to step S32 where the raw score for that query/target pair is incremented by $factor_Y$. If the user returns feedback indicating that the probable was not the target resource the user

had in mind, the flow proceeds to step S34 where the raw score for that query/target pair is decremented by $factor_N$. If the user provides no feedback, then the flow proceeds to step S36 where the raw score is decremented by $factor_O$, which can be zero. After execution of any of steps S32, S34 or S36, the flow proceeds to step S38, at which the experience level score is incremented by $Efactor_C$.

Figure 4B illustrates a preferred technique for weighting in accordance with user feedback in the case of possibles, i.e., items on the list presented to the user when no probable result can be located. As shown in the figure, if a possible is selected by the user from the presented list, at step S40, the fact of selection is recognized, preferably by use of a redirect server that allows the system to keep track of which link was chosen. Additionally, the amount of time the user spends at the selected link may be ascertained. Making use of the information gathered in the redirect and such other feedback as may be obtained, the raw score for the query/target pair is incremented, at step S44, by $factor_S$. The user is then requested to provide additional feedback after the user has finished viewing the link.

In a preferred embodiment of the present invention, this feedback is gathered from the user by presenting the user with a frame that includes a mechanism, such as a check box, or radio button, that allows the user to indicate whether the selected possible was in fact the intended or "correct" target resource. If it is

determined, at step S42, from the feedback that the link was the correct target, the flow proceeds to step S46, where the raw score for that query/target pair is incremented by $factor_{Y'}$. If the user returns a negative response, the raw score of the pair is decremented at step S48 by a by $factor_{N'}$. If no feedback is received, the raw score is decremented, at step S50, by $factor_{O'}$, which can be zero. After execution of any of steps S44, S46, S48 or S50, the flow proceeds to step S52, at which the experience level score is incremented by $Efactor_{PS}$ in the case of selection of the link, and by $Efactor_{PC}$ if the link was the correct.

Figure 5 illustrates a detail of how the present invention ranks and discriminates a probable target. At step S100 a list of possibles is obtained. Next, the list is ranked, at step S102, on the basis of the expected probability as the target. In step S104, a discrimination criteria is calculated and compared with a predetermined threshold parameter. For example, if ProbTi is the expected probability that Ti is the correct target, a formula such as the example shown can be used to determine whether T1 stands out as more probable than T2 by a relative margin that exceeds a set threshold needed to judge it as the probable intended one target. When the threshold is not exceeded, the implication is that one of the secondary possibilities may very well be the intended one, and that directing the user to the slightly favored target may not be desirable.

In the preferred embodiment, when a link on a list of possibles is selected by the user, rather than connect the user immediately to the chosen link, the finder server first redirects the user to a redirect server where feedback data relating to the selection can be gathered. One item of feedback that may be obtained in this manner is the very fact of the selection. Further feedback can be obtained by additional means, such as monitoring how long the user spends at the selected link, and by directly querying the user.

The redirect linking technique uses the target URL as a server parameter within a composite URL to control the intermediate server parameter within the URL to control the intermediate server. The target URL is embedded as a server parameter within a URL that addresses the redirect server, and the URL parameter is used to control the intermediate server process. Thus a server is called with a first URL, a redirect URL, that specifies the second URL, i.e., the target URL, as a parameter. For example

http://redirector.com/redirector?query12345678/targetserver.com/targetpath1/targetpage1.htm

where redirector.com is the intermediate server URL, query12345678 is a unique identifier of the user-query combination, and targetserver.com/targetpath1/targetpage1.htm is the target URL. The network ignores the parameter portion of the URL, which is passed as data to the server. The server acts on the parameter to perform desired

intermediary processing, in this case, the logging of the fact that this link was clicked in response to query12345678, and to redirect the user to the intended location specified by the second URL. The token query12345678 could be a unique identifier corresponding to a logged user-query entry, or it could be the actual query string.

The delay required for the redirect provides the opportunity for the display of interstitial advertisements. In addition, additional user feedback can be solicited during the delay, and the connection to the targeted URL can be aborted if the user indicates that the target site is not the one he or she intended. In addition to using the redirect when a link is selected, the technique also preferably is used when an exact match is found, to provide a brief delay before connecting the user to the exact match, to present advertisements to give the user the time to abort the connection. In any event, the user preferably is given the opportunity to provide feedback after connecting to any site, whether directly as a result of an exact match, or as a result of selecting from a linked possibles list.

The redirect server of the present invention allows data to be gathered on each link as it is followed and redirected. The redirect link can be created in a simple static HTML. However, it is preferable to create the link dynamically for each user selection.

The finder is setup to recognize the feedback function, possibly as a CGI or other gateway/API function, and invoke the appropriate function to parse the URL or other data (referer, cookies, etc.), extract the target URL and feedback information for processing, and return a page containing a redirect (or use framing or other means) to take the user to the desired target.

This mechanism is general, and can be used for many purposes. In the case of the finder server:

-Reasonably complex feedback information can be obtained, which at minimum would include the original guess. Thus a log of each guess that was not clearly resolved, paired with the corresponding user-selected target, can be obtained.

-That set of selected guess/target pairs can then be used to adjust the confidence levels in the guess/target database. Similar data on directly resolved pairs would also be applied, along with any data from wrong-match reports.

Other applications are to any situation where links go to sites other than the source. This would include results of conventional search engines, as well as resource directories, sites referring users to suppliers, advertisers, etc.

While the present invention has been discussed primarily in terms of its applicability to searching the Web, the concept has much broader applicability. For example, in the area of robot control, the above

techniques can be used to allow a robot to understand more readily the actual intent of a command.

For example, in the general case, analogous to discovery searching, the robot command may be performable in many ways, such as "direct the excess inventory out of the active holding bin," allowing the robot to find any of several allowed places to move the inventory to, and leaving some degree of ambiguity that complicates translation. In the n=1 case, or signifier mapping, more specific feedback heuristics can be utilized as described above for Web signifier searches, to assist the robot in determining the one acceptable action to be taken in response to the command such as "direct the excess inventory to the secondary holding bin."

Another example is a plant-floor robot that responds to natural-language typed or voice commands that could be told "shift the connection from the output rack from chute number 1 to chute number 2." This technique would be highly useful in highly replicated plants, such as local routing centers for a national package express network.

Yet another example would be a smart TV that is responsive to voice or typed commands that is told "turn on the Giants football game." Such a device could be linked to a central server to aid in learning to relate commands and details of current programming. The process is almost exactly as outlined for Internet searching above. Another example is a post office mail

sorter that identifies zip codes as commands for routing, based, for example, on OCR techniques or voice activation. In this case the queries would be the patterns in the optical scanner or the voice digitizer, and the correctness of hits would be tracked in any of various ways. The same process of the present invention would enable learning that would enhance the level of recognition and correct mapping to intended zip codes.

The above embodiments of the present invention have been described for purposes of illustrating how the invention may be made and used. The examples are relatively simple illustrations of the general nature of the many possible algorithms for applying feedback data that are possible. However, it should be understood that the present invention is not limited to the illustrated embodiments and that other variations and modifications of the invention and its various aspects will become apparent, after having read this disclosure, to those skilled in the art, all such variations and modifications being contemplated as falling within the scope of the invention, which is defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method of finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method for use on a finder server, having access to: (a) a database including (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the method by the user and plural previous users; and (b) a learning system structured to access and learn from information contained in the database, the method comprising:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

2. A method according to Claim 1, further comprising:

directing a computer of the user so as to enable that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

3. A method according to Claim 1, wherein a resource is determined, at the accessing step, as likely to be the intended target resource if the database

information indicates that a confidence level associated with that resource is of at least a predetermined level.

4. A method according to Claim 3, wherein if none of the indexed resources have an associated confidence level of at least the predetermined level, the method further comprises the following step:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest.

5. A method according to Claim 3, wherein the method further comprises the following steps:

in a first user interface element:

causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second user interface element:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

6. A method according to Claim 4, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

7. A method according to Claim 2, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

8. An apparatus comprising a finder server having: (a) a database including: (i) an index of resources available on a network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered in previous operations of the apparatus by a user and plural previous users; and (b) a learning system operable to access and learn from information contained in the database;

the finder server being operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

9. An apparatus according to Claim 8, wherein the finder server is further operable to:

direct a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined to be the intended target resource.

10. An apparatus according to Claim 8, wherein a resource is determined, in the accessing, to be the intended target resource if the database information

indicates that a confidence level associated with that resource is of at least a predetermined level.

11. An apparatus according to Claim 10, wherein the apparatus is operable to, if none of the indexed resources have an associated confidence level of at least the predetermined level, perform the following step:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

12. An apparatus according to Claim 10, wherein the apparatus is further operable to:

in a first user interface element:

cause the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second user interface element:

present the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

13. An apparatus according to Claim 11, wherein the apparatus is operable to, if a link has been selected, perform the following steps:

add information regarding the selection of the link to the feedback information in the database;

solicit user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

14. An apparatus according to Claim 9, the apparatus being further operable to:

solicit user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

15. A system for finding, in response to entry by a user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the system comprising:

finder server means having: (a) database means for storing an index of resources available on the network; and information regarding user feedback gathered in previous executions of the system by the user and plural previous users; and (b) learning system means for accessing and learning from information contained on the database;

receiving means for receiving a resource identity signifier from the user; and

accessing means for accessing the database means to determine which, if any, of the indexed resources is likely to be the desired target resource.

16. A system according to Claim 15, further comprising:

directing means for directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined in the access means to be the target resource.

17. A system according to Claim 15, wherein a resource is determined, by the access means, as likely to be the intended target resource if the database information indicates that a confidence level associated with that resource is of at least a predetermined level.

18. A system according to Claim 17, further comprising:

presenting means for, if none of the indexed resources have an associated confidence level of at least the predetermined level, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest.

19. A system according to Claim 17, further comprising:

means for, in a first user interface element, causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

means for, in a second user interface element, presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest

20. A system according to Claim 18, further comprising:

adding means for, if a link has been selected, adding information regarding the selection of the link to the feedback information in the database;

soliciting means for soliciting user feedback with regard to the selected link; and

means for, if the user indicates that the link is the resource intended by the resource identity signifier, updating the database information so as to increase the

confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

21. A system according to Claim 16, further comprising:

soliciting means for soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

means for, if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

22. A computer-readable storage medium storing code for causing a processor-controlled finder server having: (a) a database including (i) an index of

resources available on the network; and (ii) information regarding user feedback gathered in previous executions of the finder server by a user and plural previous users; and (b) a learning system structured to access and learn from information contained on the database, to perform a method of finding, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, among a plurality of resources located on a network comprising a plurality of interconnected computers, the method comprising:

receiving a resource identity signifier from the user; and

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource.

23. A computer-readable medium according to Claim 22, wherein the method further comprises the step of:

directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

24. A computer-readable medium according to Claim 22, wherein a resource is determined, in the accessing step, as likely to be the intended target resource if the database information indicates that a confidence level associated with that resource is of at least a predetermined level.

25. A computer-readable medium according to Claim 24, wherein if none of the indexed resources have an associated confidence level of at least the predetermined level, the method further comprises the following step:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence levels being ranked highest.

26. A computer-readable medium according to Claim 24, wherein the method further comprises the following steps:

in a first user interface element:

causing the user's computer to connect to the URL of the indexed resource having the highest confidence level; and

in a second user interface element:

presenting the user with a list of links to possible resources, the list being ordered on the basis of confidence level, the resources having the highest confidence level being ranked highest.

27. A computer-readable medium according to Claim 25, further comprising, if a link has been selected, the following steps:

adding information regarding the selection of the link to the feedback information in the database;

soliciting user feedback with regard to the selected link; and

if the user indicates that the link is the resource intended by the resource identity signifier, updating

the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the selected link, and if the user indicates that the link is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the selected link.

28. A computer-readable medium according to Claim 23, further comprising the steps of:

soliciting user feedback with regard to the resource to which the user's computer was directed in the directing step; and

if the user indicates that the resource to which his or her computer was directed is the resource intended by the resource identity signifier, updating the database information so as to increase the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which the user's computer was directed, and if the user indicates that the resource to which his or her computer was directed is not the resource intended by the resource identity signifier, updating the database information so as to decrease the confidence level associated with the mapping between the resource identity signifier and the address of the resource to which his or her computer was directed.

29. A system for finding resources on a network of interconnected computers on which a plurality of resources reside, the system comprising:

a client terminal operated by a user, the client terminal allowing the user to connect to resources located on the network; and

a finder server having:

(a) a database including: (i) an index of resources available on the network; and (ii) information regarding user feedback gathered in previous operations of the system by the user and plural previous users; and

(b) a learning system operable to access and learn from information contained in the database,

the finder server being operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by:

receiving a resource identity signifier from the user;

accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and

directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

30. A method of identifying, in response to entry by a user of an object identity signifier, a single intended object to be acted upon, the single intended object being intended by the user to uniquely correspond to the object identity signifier, among a plurality of possible objects, the method for use on a computer having: (a) a database including (i) an index of possible objects; and (ii) information regarding user feedback gathered in previous executions of the method by the user and plural previous users; and (b) a learning system structured to access and learn from information contained in the database, the method comprising:

receiving an object identity signifier from the user; and

accessing the database to determine, based upon the information in the database, which, if any, of the indexed objects is likely to be the object intended to be acted upon.

ABSTRACT OF THE DISCLOSURE

An apparatus for finding resources on a network comprises a finder server having: (a) a database including: (i) an index of resources available on network of interconnected computers on which a plurality of resources reside; and (ii) information regarding user feedback gathered from previous operations of the apparatus by a user and plural previous users; and (b) a learning system operable to access and learn from information contained on the database. The finder server is operable to locate, in response to entry by the user of a resource identity signifier, a single intended target resource intended by the user to uniquely correspond to the resource identity signifier, from among a plurality of resources located on the network, by: receiving a resource identity signifier from the user; accessing the database to determine, based on the information in the database, which, if any, of the indexed resources is likely to be the intended target resource; and directing a computer of the user so as to cause that computer to connect the user to the address of the resource, if any, determined as likely to be the intended target resource.

FITZPATRICK, CELLA, HARPER & SCINTO
30 ROCKEFELLER PLAZA
NEW YORK, NY 10112-3801
212-218-2100
FACSIMILE (212) 218-2200

WASHINGTON OFFICE
1900 K STREET, N.W.
WASHINGTON, D.C. 20006-1110
(202) 530-1010
FACSIMILE (202) 530-1055

CALIFORNIA OFFICE
650 TOWN CENTER DRIVE, SUITE 1800
COSTA MESA, CALIFORNIA 92626-1925
(714) 540-8700
FACSIMILE (714) 540-9823

WRITER'S DIRECT DIAL NUMBER

April 10, 2000

VIA FACSIMILE

Bruce G. Bernstein, Esq.
BTG International Inc.
2200 Renaissance Blvd.
Gulph Mills, PA 19406

REDACTED

Re: CUL Application
Our Ref.: 1311.1100

Dear Bruce:

REDACTED

we will prepare that application for filing in the Patent Office.

REDACTED

Sincerely yours,

Joseph W. Ragusa

Joseph W. Ragusa

cc: Richard Resiman

Date: 5/4/00
Sender: rreisman@teleshuttle.com (Richard Reisman)
To: Joseph W Ragusa, bruce.bernstein@btgusa.com, MSandonato@fchs.com
Priority: Normal

REDACTED

What is status on our filings?

-R

Richard Reisman - Teleshuttle Corporation
20 E 9 St.-14K, NY, NY 10003 - (212)-673-0225 (phone/fax) (Daytime: 646-227-7622)
Web: http://www.teleshuttle.com e-mail: rreisman@teleshuttle.com

===

REDACTED